INFORMATION ASSIMILATION IN MULTIMEDIA SURVEILLANCE SYSTEMS

PRADEEP KUMAR ATREY

NATIONAL UNIVERSITY OF SINGAPORE

2006

INFORMATION ASSIMILATION IN MULTIMEDIA SURVEILLANCE SYSTEMS

PRADEEP KUMAR ATREY

MS (Software Systems), B.I.T.S., Pilani, India

B.Tech. (Computer Science and Engineering), H.B.T.I. Kanpur, India

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF COMPUTER SCIENCE

NATIONAL UNIVERSITY OF SINGAPORE

2006

INFORMATION ASSIMILATION IN MULTIMEDIA SURVEILLANCE SYSTEMS	**PRADEEP KUMAR ATREY**	**2006**

Dedicated to the memories of

my father late Mr. Jagdish Prasad Atrey (1935-2005)

and

my father-in-law late Mr. Kamal Kant Kaushik (1947-1996)

Acknowledgements

This thesis is the result of four years of work during which I have been accompanied and supported by many people. It is now my great pleasure to take this opportunity to thank them.

After having worked as a Lecturer for more than 10 years, I was very keen to pursue full-time doctoral research. I thank the School of Computing, National University of Singapore for providing me this opportunity with financial support.

My most earnest acknowledgment must go to my advisor Prof Mohan Kankanhalli who has been instrumental in ensuring my academic, professional, financial, and moral well being ever since. I could not have imagined having a better advisor for my PhD. During the four years of my PhD, I have seen in him an excellent advisor who can bring the best out from his students, an outstanding researcher who can constructively criticize research, and a nice human being who is honest, fair and helpful to others.

I would also like to thank Prof Chang Ee-Chien for all his help and support as my co-supervisor for the initial period of my graduate studies.

I sincerely thank Prof Chua Tat-Seng and Prof Ooi Wei-Tsang for serving on my doctoral committee. Their constructive feedback and comments at various stages have been significantly useful in shaping the thesis upto completion.

My sincere thanks go out to Prof Ramesh Jain and Prof John Oommen

with whom I have collaborated during my PhD research. Their conceptual and technical insights into my thesis work have been invaluable.

Special thanks also go to Prof Frank Stephan and Prof Ooi Wei-Tsang for their help in developing the proof of the theorem given in this thesis.

There are a number of people in my everyday circle of colleagues who have enriched my professional life in various ways. I would like to thank my colleagues Vivek, Saurabh, Piyush, Rajkumar, Zhang and Ruixuan (from NUS) for their support and help at various stages of my PhD tenure. Thanks are also due to Dr Namunu for his help in audio processing, and to Vinay and Anurag (from IIT Kharagpur) for providing help in parts of the system implementation.

One of the most important persons who has been with me in every moment of my PhD tenure is my wife Manisha. I would like to thank her for the many sacrifices she has made to support me in undertaking my doctoral studies. By providing her steadfast support in hard times, she has once again shown the true affection and dedication she has always had towards me. I would also like to thank my children Akanksha and Pranjal for their perpetual love which helped me in coming out of many frustrating moments during my PhD research.

Finally, and most importantly, I would like to thank the almighty God, for it is under his grace that we live, learn and flourish.

Contents

Summary

Most multimedia surveillance and monitoring systems nowadays utilize multiple types of sensors to detect events of interest as and when they occur in the environment. However, due to the asynchrony among and diversity of sensors, information assimilation, i.e. how to combine the information obtained from asynchronous and multifarious sources, is an important and challenging research problem. Moreover, the different sensors, each of which partially helps in achieving the system goal, have dissimilar confidence levels and costs associated with them. The fact that at any instant, not all of the sensors contribute towards a system goal (e.g. event detection), brings up the issue of finding the best subset from the available set of sensors.

This thesis proposes a framework for information assimilation that addresses the issues of "when" and "how" to assimilate the information obtained from multiple sources in order to detect events in multimedia surveillance systems. The framework also addresses the issue of "what" to assimilate i.e. determining the optimal subset of sensor (streams). The proposed method adopts a hierarchical probabilistic assimilation approach and performs assimilation of information at three different levels - media stream level, atomic event level and compound event level. To detect an event, our framework uses not only the media streams available at the current instant but it also utilizes their two important properties - first, accumulated past history of whether they have been providing concurring or contradictory

evidences, and - second, the system designer's confidence in them. A compound event, which comprises of two or more atomic events, is detected by first estimating probabilistic decisions for the atomic events based on individual streams, and then by hierarchically assimilating these decisions along a timeline.

The framework also uses a dynamic programming based method that finds the optimal subset of media streams based on three different criteria; first, by maximizing the probability of the occurrence of event with a specified minimum confidence and a specified maximum cost; second, by maximizing the confidence in the subset with a specified minimum probability of the occurrence of event and a specified maximum cost; and third, by minimizing the cost of using the subset with a specified minimum probability of the occurrence of event and a specified minimum confidence. Each of these problems is proven to be NP-Complete. The proposed dynamic programming based method allows for a tradeoff among the above-mentioned three criteria, and offers the flexibility to compare whether any one set of media streams of low cost would be better than any other set of media streams of higher cost, or any one set of media streams of high confidence would be better than any other set of media streams of low confidence. To show the utility of our framework, we provide experimental results for event detection in a surveillance scenario.

List of Tables

List of Figures

List of Symbols

$area$	Area of the blob
ACC	Performance metric - Accuracy of event classification
A to **Z**, **a**, **b**	Used to denote optimal subset in the graph obtained using **MaxGoal**, **MaxConf**, and **MinCost** algorithms
A_{11} to A_{22}	Audio streams
$\mathcal{A}1$ to $\mathcal{A}5$	Assumptions 1 to 5
c_i	Cost per unit time of using i^{th} stream
C_n	Total cost of n streams
C_{spec}	Specified maximum overall cost
$Conf(i, m)$	Optimal confidence in the group of streams 1 to i with the cost m
$Cost(i, m)$	Optimal cost of using streams 1 to i with the probability m
C_Φ	is the cost of using the subset Φ of streams
CFusion	Confidence fusion function used in **MaxGoal**, **MaxConf**, and **MinCost** algorithms
d	Degree of precision in considering the probability value, used in Lemma 4.2.3
$\mathbf{e}_j$	j^{th} atomic event
$\mathbf{E}_k$	k^{th} compound event

Ex, Ey	Mapped location of the blob on earth
ED_{ji}	Event Detector employed to independently detect each atomic event $\mathbf{e}_j$ based on stream M_i
E	Set of events
f_i	Confidence in i^{th} stream
$f_{ii'}$	Confidence in a group of two streams M_i and $M_{i'}$
$\mathbf{F}$	Set denoting the confidence values in streams of the set $\mathbf{M}^n$
F_i, F_{i-1}	Overall confidence in a group of i and $i-1$ streams, respectively
F_{S_1}, F_{S_2}	Overall confidence in subsets S_1 and S_2, respectively
F_{spec}	Specified minimum overall confidence.
F_Φ	Overall confidence when the subset Φ of streams is used
FRR, FAR	False Rejection Ratio and False Acceptance Ratio in event classification, respectively
$\mathcal{H}1$ to $\mathcal{H}3$	Three heuristic used for obtaining the optimal subset of streams
h	Height of the blob
i	Index for the media streams
j	Index for the atomic event
k	Index for the compound event
kk	Index for the Select array used in **MaxGoal**, **MaxConf**, and **MinCost** algorithms
K	An instance of 0-1 Knapsack problem
l	Temporary array used in **MinCost** algorithm
L	Number of discrete values used for probability of the occurrence of event

m, m'	Indices used for column in computing the dynamic programming table in **MaxGoal**, **MaxConf**, and **MinCost** algorithms
M_i	i^{th} media stream
$M_{i,t}$	i^{th} media stream at time instant t
$\mathbf{M}^n$	A set of n media streams
MSP_i	A set of media processing tools for i^{th} stream
$\mathcal{M}1$ - $\mathcal{M}5$	Used in the model of computation given in the problem formulation
n	Number of sensors in the system $\mathbf{S}$
n'	Number of possible subsets satisfying the required criteria
N_a	Total number of atomic events
N_c	Total number of compound events
N_E	Total number of events
O	Big Oh notation to represent the complexity of an algorithm
$OptProb$	Temporary variable used in **MinCost** algorithm
p_i	probability of the occurrence of an event based on stream M_i
$p_i(t)$	probability of the occurrence of an event based on stream M_i at time t
$p_{j,i} = P(\mathbf{e}_j \vert M_i)$	Probability of the occurrence of atomic event $\mathbf{e}_j$ based on stream M_i
$p_{\mathbf{E}_k}$	Probability of the occurrence of compound event $\mathbf{E}_k$
$p_{\mathbf{e}_j}$	Probability of the occurrence of atomic event $\mathbf{e}_j$
$\mathbf{P}$	Set of probabilities of the occurrence of event based on streams in set $\mathbf{M}^n$

$P_{i-1} = P(\mathbf{e}_{j_t}\|\mathbf{M}_t^{i-1})$	Probability of the occurrence of atomic event $\mathbf{e}_j$ at time t based on streams M_1, M_2, ..., M_{i-1}
$P_i = P(\mathbf{e}_{j_t}\|\mathbf{M}_t^{i})$	Probability of the occurrence of atomic event $\mathbf{e}_j$ at time t based on streams M_1, M_2, ..., M_i
$\mathcal{P}(\mathbf{M}^n)$	Power set of a set $\mathbf{M}^n$ of streams
P_{spec}	Specified minimum fused probability of the occurrence of event
P_Φ	Fused probability of the occurrence of event based on a subset Φ
$P(\mathbf{e}_j\|S_1)$	Probability of the occurrence of atomic event $\mathbf{e}_j$ based on subset S_1 of streams
$P(\bar{\mathbf{e}_j}\|S_2)$	Probability of the non-occurrence of atomic event $\mathbf{e}_j$ based on subset S_2 of streams
$Prob(i, m)$	Probability of the occurrence of event based on streams 1 to i using the cost m
PFusion	Probability assimilation function used in **MaxGoal**, **MaxConf**, and **MinCost** algorithms
r	Number of atomic events in a compound event
R, R'	Temporary variables used in **MinCost** algorithm
s	Index for the media streams
ss	Index for the array l used in **MinCost** algorithm
S_1, S_2	Two subsets of streams, in favor and in against the occurrence of event

Select	Array used in **MinCost** algorithm
S	Multimedia Surveillance System
t_i	Minimum time interval in which decisions about an event are obtained
t_w	The time interval in which the streams should be assimilated
T_c	Function used to denote the consensus rule
T_r	Transformation function used to map an instance of 0-1 Knapsack problem into an instance of Media Selection problem
Th	Threshold used for the probability of the occurrence of event
u_i	i^{th} item in the 0-1 KNAPSACK problem
$\mathbf{U}^n$	Set of items in the 0-1 KNAPSACK problem
V_{11} to V_{22}	Video streams
w	Width of the blob
w_i'	Weight assigned to i^{th} media stream using a consensus rule
w_i	Weight of i^{th} item in the 0-1 KNAPSACK problem
W	Set denoting the weights of items in the 0-1 KNAPSACK problem
W_{spec}	Knapsack capacity in the 0-1 KNAPSACK problem
W_Λ	Total weight of items of subset Λ in the 0-1 Knapsack problem
x, y	Image coordinates of the blob
x_i	Profit from i^{th} item in the 0-1 KNAPSACK problem
X	Set denoting the profits from items in the 0-1 KNAPSACK problem
X_{spec}	Minimum specified profit in the 0-1 KNAPSACK problem
X_Λ	Total profit from a subset Λ of items in the 0-1 Knapsack problem

α_i	Normalization factor for integrating i^{th} stream into the assimilation process
$\overline{\gamma}_i$	Agreement coefficient between two sources M^{i-1} and M_i
$\gamma_{ii'}(t)$	Agreement coefficient between M_i and $M_{i'}$ at time instant t
ρ, ρ'	Used for replacing P_{i-1} for simplification in Lemma 4.2.3
σ, σ'	Used for replacing p_i for simplification in Lemma 4.2.3
$\Gamma(t)$	A set of agreement coefficients at time instant t
Φ	Optimal subset of media streams in a Media Selection problem
Λ	Optimal subset of items in the 0-1 Knapsack problem

Chapter 1

Introduction

Security has been a driving impetus for civilization for several centuries. Recent increase in terrorist activities across the globe has forced governments to make public security an important part of their policy. In turn, a majority of developed cities around the world are now being equipped with the current-generation automated surveillance systems [83] that consist of thousands of multiple types of sensors including video cameras and even microphones with a primary goal of automatically detecting and recording the events of interest as and when they occur.

In recent times, it is also being increasingly accepted that most surveillance and monitoring tasks can be better performed by using multiple types of sensors as compared to using only a single type. This is because a single type of sensors can only partially help in accomplishing surveillance tasks due to their ability to sense only a part of the environment. Moreover, the multiple types of sensors capture different aspects of the environment to provide complementary information which is not available from a single type. Therefore, the surveillance systems nowadays more often utilize multiple types of sensors like microphones, motion detectors and RFIDs etc in

addition to the video cameras.

In multimedia surveillance and monitoring systems, where a number of asynchronous heterogeneous sensors are employed, the assimilation of information obtained from them in order to accomplish a task (e.g. event detection) is an important and challenging research problem. *Information assimilation refers to the process of combining the sensory and non-sensory information using the context and the past experience.* The issue of information assimilation is important because the assimilated information obtained from multiple sources provides more accurate state of the environment than the individual sources. It is challenging because the different sensors provide the correlated sensed data (we call it "stream" from here onwards) in different formats and at different rates. For example, a video may be captured at a frame rate which could be different from the rate at which audio samples are obtained, or even two video sources can have different frames rates. Moreover, the processing time of different types of data is also different. Also, the designer of a system can have different confidence levels in different sensors while detecting different events.

Event detection is one of the fundamental analysis tasks in multimedia surveillance and monitoring systems. This thesis proposes an information assimilation framework for event detection in multimedia surveillance and monitoring systems.

Events are usually not impulse phenomena in real world, but they occur over an interval of time. Based on different granularity levels in time, location, number of objects and their activities, an event can be a "compound event" or simply an "atomic event". This representation of events is similar to [12, 60], however, our basis of categorization is different. We define compound events and the atomic events as follows.

Definition 1 *Event is a physical reality that consists of one or more living or non-living real world objects (who) having one or more attributes (of type) being involved in one or more activities (what) at a location (where) over a period of time (when).*

Definition 2 *Atomic event is an event in which exactly one object having one or more attributes is involved in exactly one activity.*

Definition 3 *Compound event is the composition of two or more different atomic events.*

A compound event, e.g. "a person is running and shouting in the corridor" can be decomposed into its constituent atomic events - "a person is running in the corridor" and "a person is shouting in the corridor". The atomic events in a compound event can occur simultaneously, as in the example given above; or they may also occur one after another, e.g. the compound event "A person walked through the corridor, stood near the meeting room, and then ran to the other side of the corridor" consists of three atomic events "a person walked through the corridor" followed by "person stood near the meeting room", and then followed by "person ran to the other side of the corridor".

The different atomic events, to be detected, may require different types of sensors. For example, a "walking" and "running" event can be detected based on both video and audio streams, whereas a "standing" event can be detected by using video streams but not by using audio streams, and a "shouting" event can be better detected using the audio streams. Since an atomic event can be detected based on more than one media streams, the atomicity of an event cannot be defined at the sensor level. The different atomic events require different minimum time periods over which they can be

confirmed. This minimum time period for different atomic events depends upon the time in which the amount of data sufficient to reliably detect an event can be obtained and processed. Even the same atomic event can be confirmed in different time periods using different data streams. For example, minimum video data required to detect a walking event could be of two seconds, while the same event can be detected based on audio data of one second.

1.1 Issues in Information Assimilation

The media streams in multimedia surveillance and monitoring systems, in general, have the following characteristics - first, they are often correlated; second, the system designer has different confidence levels in the decisions obtained based on them; and third, there is a cost of obtaining these decisions which usually includes the cost of sensor, its installation and maintenance cost, the cost of energy to operate it, and the processing cost of the stream. We assume that each stream in a multimedia surveillance and monitoring system partially helps in detecting an event.

The various research issues in the assimilation of information in such systems are as follows:

1. *When to assimilate?.* Events occur over a timeline [22]. Timeline refers to a measurable span of time with information denoted at designated points. Timeline-based event detection in multimedia surveillance systems requires identification of the designated points along a timeline at which assimilation of information should take place. Identification of these designated points is challenging because of asynchrony and diversity among streams and also because of the fact that different

events have different granularity levels in time.

2. *What to assimilate?* The fact that at any instant all of the employed media streams do not necessarily contribute towards accomplishing the analysis task (e.g. detection of an event) brings up the issue of finding the most informative subset of streams. From the available set of streams,

    - What is the optimal number of streams required to detect an event under the specified constraints?
    - Which subset of the streams is the optimal one?
    - In case the most suitable subset is unavailable, can one use alternate streams without much loss of cost-effectiveness and confidence?
    - How frequently should this optimal subset be computed so that the overall cost of the system is minimized?

3. *How to assimilate?* In combining of different streams,

    - How to utilize the correlation among them?
    - How to integrate the contextual information (such as environment information) and the past experience?

1.2 Proposed Framework: Characteristics

The proposed information assimilation framework addresses the above-mentioned issues and has the following distinct characteristics -

- *Late thresholding over early thresholding*: The detection of events based on individual streams is usually accomplished with uncertainty.

To obtain a binary decision, early thresholding of uncertain information about an event may lead to error. For example, let an event detector find the probabilities of the occurrence of an event based on three media streams M_1, M_2 and M_3, to be 0.60, 0.62 and 0.70, respectively. If the threshold is 0.65, then these probabilistic decisions are converted into binary decisions 0, 0 and 1, respectively; which implies that the event is found occurring based on stream M_3 but is found non-occurring based on stream M_1 and M_2. Since two decisions are in favor of the non-occurrence of event compared to the one decision in favor of the occurrence of event, by adopting a simple voting strategy, the overall decision would be that the event did not occur. It is important to note that early thresholding can introduce errors in the overall decision. In contrast to early thresholding, the proposed framework advocates late thresholding by first assimilating the probabilistic decisions that are obtained based on individual streams, and then by thresholding the overall probability (which is usually more than the individual probabilities, e.g. 0.85 in this case) of the occurrence of event based on all the streams, which is less erroneous.

- *Use of agreement/disagreement among streams*: The sensors capturing the same environment usually provide concurring or contradictory evidences about what is happening in the environment. The proposed framework utilizes this agreement/disagreement information among the media streams to strengthen the overall decision about the events happening in the environment. For example, if two sensors have been providing concurring evidences in the past, it makes sense to give more weight to their current combined evidence compared to the case if they provided contradictory evidences in the past [73]. The agree-

ment/disagreement information (we call it as "agreement coefficient") among media streams is computed based on how similar or contradictory decisions have been made using them in the past. We also propose a method for fusing the agreement coefficients among the media streams.

- *Use of confidence in streams*: The designer of a multimedia surveillance system can have different confidence levels in different media streams for detecting different events. The proposed framework utilizes the confidence information by assigning a higher weight to the media stream which has a higher confidence level. The confidence in each stream is computed based on how accurate it has been in the past. Integrating confidence information in the assimilation process also requires the computation of the overall confidence in a group of streams, a method for which is also proposed.

- *Dynamic programming approach for optimal subset selection*: The proposed framework adopts a dynamic programming approach that finds the optimal subset of media streams so as to achieve the surveillance goal under specified constraints. It finds the optimal subset of media streams based on three different criterion:

    1. By maximizing the probability of achieving the surveillance goal (e.g. event detection) under the specified cost and the specified confidence.

    2. By maximizing the confidence in the achieved goal under the specified cost and the specified probability with which the surveillance goal is achieved.

    3. By minimizing the cost to achieve the surveillance goal with a

specified probability and a specified confidence.

Each of these problems is proven to be NP-Complete. The proposed approach also allows for a tradeoff among the above-mentioned three criteria, and offers a flexibility to compare whether any one set of media streams of low cost would be better than any other set of media streams of higher cost, or any one set of media streams of high confidence would be better than any other set of media streams of low confidence.

- *Information assimilation over information fusion*: Information assimilation is different from information fusion in that the former brings the notion of integrating context and the past experience in the fusion process. The context is an accessory information that helps in the correct interpretation of the observed data. The proposed framework uses the geometry of the monitored space along with the location, orientation and coverage area of the employed sensors as the spatial contextual information. It integrates the past experience by modeling the agreement/disagreement information among the media streams based on the accumulated past history of their agreement or disagreement.

1.3 Thesis Contributions

The main contributions of this thesis are as follows.

- This thesis proposes a framework for assimilation of information in order to detect events in surveillance and monitoring systems. The framework introduces the notion of compound and atomic events that helps in describing events over a timeline. The proposed framework, in the assimilation process, utilizes two distinct properties of sensors

- the agreement/disagreement information among and the confidences in them.

- The thesis presents a NP-Completeness proof for the problem of optimal subset selection of streams, and also proposes a near-optimal solution to it using a dynamic programming based method. The dynamic programming based approach allows for a tradeoff between extent to which a surveillance goal is achieved using the optimal subset, the cost of using the optimal subset, and the confidence in the optimal subset of streams. The approach also offers the user a flexibility to choose alternative (or the next best) subset when the best subset is unavailable.

1.4 Thesis Organization

This thesis is organized as follows. In Chapter 2, we present a review of the fundamental methods used in past for information fusion and for optimal sensor selection. It is discussed how information assimilation can be performed by integrating into information fusion process the various properties of information obtained from different sources. The existing approaches for fusion of multimodal information adopted by multimedia researchers are described and a categorization of the existing fusion approaches is provided. We also describe the past works related to multimodal information fusion at different levels such as feature-level (early fusion) and decision-level (late fusion). This chapter has also provided a review of the past works on using the measures of correlation, confidence information and the contextual information. Finally, we also present the past approaches for optimal subset selection of streams.

Chapter 3 presents the proposed information assimilation framework for event detection in multimedia surveillance and monitoring systems. In this chapter, we first formulate the problem of information assimilation in the context of multimedia surveillance, and then describe how the framework addresses the issues of "when" and "how" to assimilate the information obtained from multiple sources. The significance of timeline in event detection is discussed and a hierarchical probabilistic method used for information assimilation is presented in greater detail. Simulation results are also presented to show the effect of using agreement/disagreement information in the assimilation process.

In Chapter 4, we describe how the proposed framework addresses the issue of "what to assimilate" in order to accomplish a surveillance task. For determining the optimal subset of streams in order to detect events in surveillance and monitoring systems, three different Multimedia Selection problems are first introduced and then are proved to be NP-Complete. The dynamic programming based solutions to these three different problems are presented with a discussion on their time and space complexities. The chapter concludes with simulation results (on synthetic data) that show the utility of dynamic programming based method.

To demonstrate the utility of the proposed framework, the experimental results on real data are presented in Chapter 5. This chapter begins with a brief description of the surveillance system which we have implemented. Then, the results for information assimilation and for optimal subset selection are provided. It is also established that the use of agreement/disagreement information among streams and the use of confidence information in streams helps in better detection of events in surveillance environment.

Chapter 6 presents summary and conclusions of this dissertation. This dissertation shows how the proposed information assimilation framework is useful for event detection in a multimedia surveillance environment. However, the application of this framework in other context is an issue which needs to be explored in future research. Also, there are several other research issues which are out of scope of thesis and which open up a wide spectrum of topics for future research. This is the point of discussion in Chapter 6 on future research directions.

Chapter 2

Related Work

As the focus of this thesis is on information assimilation, this chapter presents a brief review of some of the fundamental concepts and ideas related to it that has been proposed in the existing literature. As discussed earlier, information assimilation is different from information fusion in that the former brings the notion of contextual information and past experience. In this chapter, we present the past works related to information fusion, and we also discuss how information assimilation can be performed by integrating into information fusion process the various properties of the information obtained from different sources.

A significant amount of work has been done by multimedia (including computer vision) researchers in the context of video surveillance, such as for face detection [87, 38], moving object detection [44], object tracking [19], object classification [24], [44], human behavior analysis [61], people counting [91], and abandoned object detection [76, 74]. Valera and Velastin [83] have recently presented a survey on the state of the art of surveillance systems.

A few works have also been reported for the surveillance using audio. The examples of various audio events detected in the past include glass

breaks, explosions or door alarms [27], talking person, falling chair [25], impulsive gun shots [23], human's coughing in the office environment [34] and the working of an air-conditioner [56].

This thesis does not aim to review the works which are specific to video surveillance or audio surveillance. Since the focus of thesis is on surveillance using multiple media, we provide in this chapter a literature survey of the works which include more than one medium.

This chapter is organized as follows. In section 2.1, we first present a broad categorization (Probabilistic and Non-probabilistic methods) of traditional multimodal information fusion techniques; and then, we describe the past works related to multimodal information fusion at different levels such as feature-level (early fusion) and decision-level (late fusion). Section 2.2 describes the use of agreement/disagreement information in the past works, and section 2.3 elaborates on how the confidence information has been used in multisensor systems. The past works related to using contextual information is described in section 2.4. Finally, we present the past approaches for optimal subset selection of streams in section 2.5.

2.1 Multi-modal Information Fusion Methods

Multimodal information fusion refers to combining information from multiple modes. The information could be sensory (such as from audio and/or video sensors) or non-sensory (such as from world wide web and/or database etc). In general, the integration of different modes of information can be achieved at two levels [33] - *Feature-level fusion* (or *early fusion*) and *Decision-level fusion* (or *late fusion*) as shown in figure 2.1. In early fusion, the features ($Feature_1$ to $Feature_n$) extracted from sensor data are first combined and then input to a single event detector (ED) that eventually

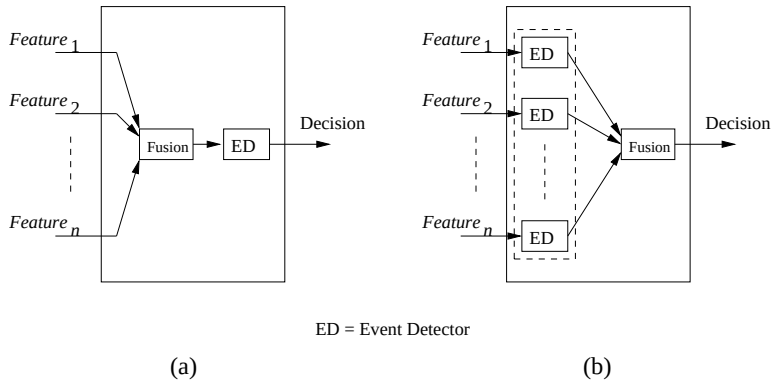

Figure 2.1: Fusion strategies: (a) Early fusion (b) Late fusion

provide the decision about an event. On the other hand, in late fusion, the event detectors (ED_1 to ED_n) first provide the local decisions that are obtained based on individual features ($Feature_1$ to $Feature_n$); and then these local decisions are combined to make a global decision.

The following subsections are organized as follows. In subsection 2.1.1, we first present various traditional fusion strategies reported in literature; and then in subsections 2.1.2 and 2.1.3, we describe how these fusion strategies have been adopted by researchers for a variety of applications at feature level and decision level, respectively.

2.1.1 Traditional information fusion techniques

Information fusion is a well developed research area. In context of multimedia also, the researchers have used various fusion methodologies. Luo et al. [54] provided a classification of sensor fusion methods as shown in figure 2.2. Their proposed classification is valid except that there could be some overlap in different categories. For example, the classification methods such as Hidden Markov model, Gaussian mixture model etc can also be put into the inference methods category. Similarly, the fusion method based on Self

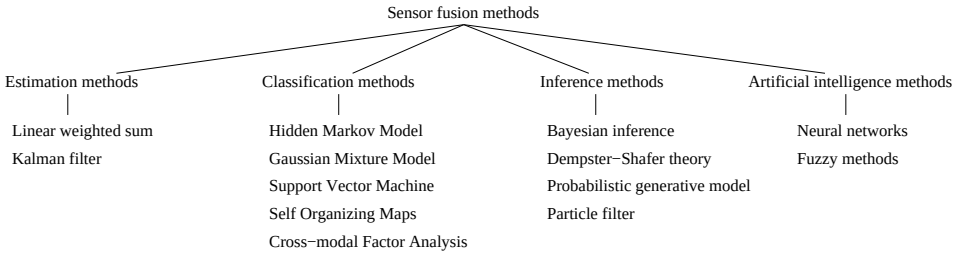

Figure 2.2: A classification of sensor fusion methods proposed by Luo et al. [54]

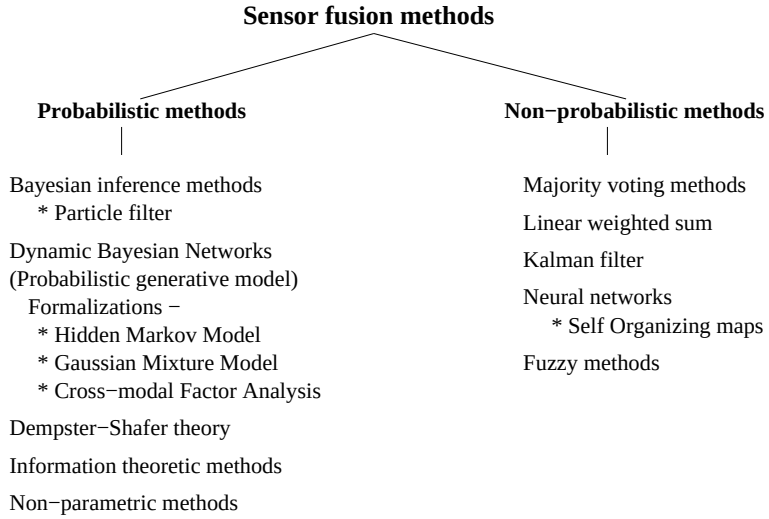

Figure 2.3: Our proposed classification of sensor fusion methods

Organizing Maps adopts the principle of neural networks. Also, Bayesian inference method can be used for classification.

In order to remove these ambiguities in this classification, we propose a new classification by grouping the sensor fusion methods into the following two broad categorizes (as shown in figure 2.3):

- Probabilistic fusion methods
- Non-probabilistic fusion methods

Probabilistic fusion methods

The probabilistic fusion methods are based on first learning the joint distributions of data and then inferring from it the posterior probability of a hypothesis being true. The commonly used methods in this group are: Bayesian inference method, Dynamic Bayesian Networks, Dempster-Shafer method, Information theoretic models and Non-parametric methods. We briefly introduce these methods in the subsequent paragraphs.

Bayesian inference methods are often referred as the 'classical' or 'canonical' sensor fusion methods because not only are they the most widely used, but also they are the basis of, or the starting points for, many new methods [33]. Bayesian inference method quantitatively computes the joint probability (by using the *product rule*) that the observations obtained from multiple sensors can be attributed to a given assumed hypothesis but it lacks in ability to handle mutually exclusive hypotheses and general uncertainty.

Dynamic Bayesian Networks (DBN) are directed graphical models of stochastic processes in which the hidden states are represented in terms of individual variables or factors. A DBN is specified by a directed acyclic graph, which represents the conditional independence assumption and the conditional probability distributions of each node [45]. With the DBN representation, the classification of the decision fusion models can be seen in terms of independence assumptions of the transition probabilities and of the conditional likelihood of the observed and hidden nodes. A variation of Dynamic Bayesian Networks is the *probabilistic generative model* that ensures the Bayes optimality and utilizes the temporal dynamics while maintaining the optimality properties [35]. The various formalization of graphical models include Hidden Markov Models (HMM), Gaussian Mixture Models (GMM) and Cross-modal Factor Analysis (CFA).

The *Dempster-Shafer* method generalizes Bayesian theory to relax the Bayesian method's restriction on mutually exclusive hypotheses, so that it is able to assign evidence to the unions of hypotheses [88].

Information theoretic methods are based on computing mutual information and entropy between sensor data. Mutual information quantifies the information that two random variables convey about each other [29]. Mutual information between two data sources is computed by assuming them to jointly follow the Gaussian distribution [36]. Entropy based model constructs an exponential function that fuses multiple features to approximate the posterior probability of an hypothesis given the data [37].

In contrast to above probabilistic methods, which assume the multimodal data to locally and jointly follow any specific distribution (usually Gaussian), the *Non-parametric probabilistic* methods do not assume any specific distribution in combining of data and statistically estimate the parameters [29].

Non-probabilistic fusion methods

Non-probabilistic methods use the absolute data (feature or decision) values for combining them. The common used methods in this category include Majority voting, Linear weighted sum, Kalman filter, Neural networks methods, and Fuzzy methods. They are briefly described as follows.

Majority voting sensor fusion imitates voting as a means for human decision-making. It combines detection and classification declarations from multiple sensors by treating each sensors declaration as a vote, and the voting process may use majority, plurality, or decision-tree rules ([49] Chapter 7).

A variation of majority voting method is the *Linear weighted sum* method,

which uses a linear combination fusion strategy by assigning the normalized weights to different sensor data streams [86]. This method has widely been adopted in multimedia analysis research. In contrast to a weighted average, Kalman filter is predominantly preferred because it provides better estimates for the fused data that are optimal in a statistical sense [54].

Neural networks methods consist of a network of nodes. The input nodes accept sensors output data, and the output nodes show sensor fusion results. The input nodes are connected to output nodes via interconnecting data paths. The weights along these data paths decide the input-output mapping behavior, and they can be adjusted to achieve desired behavior. This weight-adjusting process is called training, which is realized by using a large number of input-output pairs as examples [15]. A formalization of Neural networks method is Self Organizing Maps [31].

Fuzzy logic methods accommodate imprecise states and variables. It provides tools to deal with observations that is not easily separated into discrete segments and is difficult to model with conventional mathematical or rule-based schemes [88].

Other non-probabilistic statistical methods such as *Max rule* and *Min rule* approximate the fused value based on maximum and minimum of the sensor data values, respectively. Since these methods are biased towards maximum or minimum of the data and do not represent the true fused value, hence are usually not applicable.

After the brief introduction of traditional sensor fusion methods, in the next two subsections, we describe how these fusion approaches have been adopted by the researchers at feature-level and at decision-level.

2.1.2 Feature-level multi-modal fusion

Researchers have used early fusion strategy to perform the *audio-visual* fusion for solving diverse problems including speech processing [35] and recognition [58], monologue detection [62, 40], audio-video localization [36, 29, 52] and speaker tracking [70, 20].

Hershey et al. [35] proposed to use a probabilistic generative model to combine audio and video by learning the dependencies between the noisy speech signal from a single microphone and the fine-scale appearance and location of the lips during speech. In the other work, Hershey and Movellan [36] obtained generic measures of 'audio-visual synchrony' by defining random variables related to the audio and video signals, and then evaluates the correlation or mutual information (MI) relationships between those random variables. In both the works, the authors assume that audio and video signals are individually and jointly Gaussian random variables.

Nock et al. [62] extended the approach proposed in [36] for monologue detection by relaxing the single Gaussian assumption and allowing the audio and video signals to be locally Gaussian. They introduced two techniques as VQ-based MI and Gaussian-based MI respectively. With either scheme, the face amongst a set of possibilities that is deemed to have produced a given audio sequence provides the highest mutual information score.

In contrast to the above approaches, where audio and video are assumed to locally and jointly follow Gaussian distribution, Fisher-III et al. [29] presented a non-parametric approach to learn the joint distribution of audio and visual features. They estimated a linear projection onto low-dimensional subspaces to maximize the mutual information between the mapped random variables. The approach is used for audio-video localization.

Nefian et al. [58] used the statistical property of coupled Hidden Markov

Model (HMM) to model the state asynchrony of the audio and visual observation sequences while preserving their correlation over time. The approach is used for speech recognition. Iyengar et al. [40] adopted a weighted linear sum fusion approach for monologue detection using face, speech and the synchrony score between them. Later, the authors extended their approach for semantic concept detection and annotation in video [41]. Li et al. [52] investigates different cross-modal association methods using the linear correlation model, and present a method called Cross-modal Factor Analysis (CFA) that uses the cross-modal association. They show its applicability to information retrieval and to detect talking heads.

Audio-visual modalities have also been used for speaker tracking. Perez et al. [70] presented a method that fuses 2-D object shape and audio information via importance filters. They used audio information to generate an importance sampling function, which guides the random search process of particle filter towards regions of the configuration space likely to contain the true configuration (a speaker). Checka et al. [20] formulates the multiple person tracking problem using a state estimation framework. They applied a particle filter with audio and video state components, and derive observation likelihoods based on both audio and video measurements. The state includes the number of people present, their positions, and whether each person is talking.

Other approaches which adopted feature-level audio-video fusion approach include Beal et al. [10] for object tracking, Wang et al. [84] for face tracking, Wang et al. [86] for face detection and activity monitoring. A summary of all the fusion approaches adopted at different levels is given in Table 2.1.

It is observed that only the works [36], [29], [62], [58], [52], [10] and [84]

utilized the feature-level correlation among multiple modalities in different forms such as mutual information and cross-correlation coefficient. The other works [35], [40], [41], [70], [20] and [86] did not make explicit use of the feature-level correlation among different modalities.

In addition to the video and audio, other modalities such as *closed-caption text*, *external metadata* have also been used for several applications such as video indexing and the content analysis for team sports video. Babaguchi et al. [8] present a method for event based indexing of sports video using inter-modal collaborations. In [9], they extended it for highlight extraction based on *sound cues* and *gamestats* (from some websites).

Chaisorn et al. [18] also presented a HMM-based multi-modal approach for news video story segmentation by using a combination of features include *visual-based features* such as color, *object-based features* such as face, video-text, *temporal features* such as audio and motion, and *semantic feature* such as cue-phrases.

The fusion strategies adopted by Babaguchi et al. [8] and Chaisorn et al. [18] are suitable in the context of news and sports video analysis. However, they do not follow any formal model of fusion which is essential for assimilating sensor information in multimedia surveillance systems.

Gandetto et al. [31] presented an architecture for multisensor data fusion in the context of Ambient Intelligence (AmI). The proposed system integrated an heterogeneous network of sensors with CCD cameras and computational units working together in a LAN. A Self Organizing Map (SOM) based method is used to classify the events into different categories.

Wu [88] proposed to use Dempster-Shafer theory for sensor fusion in the context of context-aware computing, and also discussed the relationship between classical Bayesian method and Dempster-Shafer theory.

In the following subsection, we describe decision-level fusion approaches that has been reported in literature.

2.1.3 Decision-level multi-modal fusion

Several conventional information fusion methods have been used to perform fusion at decision-level with the basic assumption of independence among information sources [72]. Such an assumption does not hold for the real world data fusion applications, for example, in the assimilation of streams in multimedia surveillance systems where we obtain correlated decisions based on data captured from different sensors. It necessitates the use of more sophisticated algorithms which take into consideration of how the decisions obtained based on various streams co-vary with each other. In the subsequent paragraphs, we restrict our discussion to the methods for fusion of correlated decisions.

Chair and Varshney [17] established an optimal fusion rule with the assumption that each local sensor made a predetermined decision and each observation was independent. Kam et al. [48] generalizes their solution for fusing the correlated local decisions. The major drawback of their methods is the requirement of the knowledge of a priori probabilities and the probabilities of a miss and detection of each local sensor that are not readily available in practice. Chen and Ansari [21] derived another form of the maximum posterior probability (MAP)-based optimal fusion rule. In their algorithm, they express the log-likelihood ratio function as a linear combination of ratios of conditional probabilities and local decisions. The estimations of the conditional probabilities are adapted by reinforcement learning. O'Brien [63] presented a method for fusion of correlated probabilities where each probability value corresponds to a local decision. The author incorporates

the correlation between the decisions by assuming conditional independence between some function of individual probabilities. However, the basis on which the author choose this function is not obvious. Rao and Whyte [72] also proposed a Bayesian inference method for identification of target object in decentralized multisensor system. Our Bayesian formulation is similar to [72], however, we have also incorporated the agreement/disagreement information among and the confidence in sensors.

From the multimedia point of view, decision fusion based *audio-visual* observations is applied for digit recognition by Meyer et al. [57]. The authors have, however, assumed the conditional independence of audio and visual features to multiply the a posteriori probabilities for the audio and visual data streams. Hsu et al. [37] used a Maximum Entropy model that constructs an linear exponential function that fuses multiple local binary decisions (derived based on various media streams) to approximate the posterior probability of an event. They used raw multi-modal features like *Anchor face*, *Commercial*, *Pitch jump*, *Significant pause*, *Speech segments and rapidity* etc for the purpose of story segmentation in news videos. Neti et al. [59] presented an audio-visual approach for multimedia indexing and human-computer interaction. They employed a linear weighted sum fusion strategy to combine the decisions obtained based on different audio-visual cues. Stauffer [78] presented an audio-visual based method for the automatic clustering and for learning the salient temporal relationship between audio and visual events by introducing a concept of casual link analysis between the events (i.e. at decision level). However, the focus of this work is away from fusion.

Table 2.1: A summary of multi-modal fusion methods

The work	Level of fusion	Measure of correlation	Fusion method used
Hershey et al. [35]		-	Probabilistic generative model
Hershey and Movellan [36], Nock et al. [62]	Feature	Mutual information	Probabilistic generative model
Beal et al. [10]		Cross correlation	Probabilistic generative model
Fisher-III et al. [29]	Feature	Mutual information	Non-parametric model
Nefian et al. [58]	Feature	Cross correlation	Hidden Markov Model
Iyenger et al. [40], [41] Wang et al. [86]	Feature	-	Linear Weighted Sum
Li et al. [52]	Feature	Cross correlation	Cross-modal factor analysis
Perez et al. [70], Checka et al. [20]	Feature	-	Probabilistic model (Particle Filter)
Stauffer [78]	Decision	Casual link analysis	No formal fusion
Wang et al. [84]	Feature	Cross correlation	Kalman filter
Babaguchi et al. [8], [9]	Feature	-	No formal fusion
Chaisorn et al. [18]	Feature	-	Hidden Markov Model
Gandetto et al. [31]	Feature	-	Self organizing maps
Wu [88]	Feature	-	Dempster-Shafer theory
Chair and Varshney [17]	Decision	-	Bayesian inference model
Kam et al. [48] Chen and Ansari [21]	Decision	Cross Correlation	Bayesian inference model
O'Brien [63] Meyer et al. [57]	Decision	-	Bayesian inference model
Hsu et al. [37]	Decision	-	Maximum entropy model
Neti et al. [59]	Decision	-	Linear weighted sum
Rao and Whyte [72]	Decision	-	Bayesian inference model
Feng et al. [28]	Both	-	Support Vector Machine
Wu et al. [89]	Both	Cross[$] correlation	Linear weighted sum
Xu and Chua [90]	Both	-	Hidden Markov Model, Rule based, Linear weighted sum, Bayesian inference model
Our approach	Both*	Cross correlation and Agreement coefficient[!]	Bayesian inference model

'-' Indicates that the authors have not explicitly used the measure of correlation.
*The proposed approach employs early (feature level) assimilation at intra-media stream level and late (decision level) assimilation strategy at inter-media stream level.
[$] The cross-correlation between features at both intra-media stream level as well as inter-media stream level.
[!]The cross-correlation and the agreement coefficient are used as a measure of correlation at intra-media stream level and at inter-media stream level, respectively.

2.1.4 The hybrid approach for assimilation

The feature-level fusion approaches described in section 2.1.2 have demonstrated that the multimedia researchers have widely used them for various applications. However, the feature-level fusion of information has certain limitations. In a multimedia surveillance environment, where several different types of sensors are used, the number of modalities significantly increases and consequently it becomes difficult to learn the cross-correlation among them. On the other hand, the decision-level fusion approach fails to utilize the feature-level correlation among the different modalities (e.g. color, edge etc.) of the same medium (e.g. image). Therefore a multi-level (or hybrid) approach, in which fusion takes place at feature as well as decision level, may be more appropriate.

Wu et al. [89] proposed a two-step fusion approach. The first step finds statistically independent modalities from raw features (feature level fusion). In the second step, we use super-kernel fusion to determine the optimal combination of individual modalities (decision level fusion). The authors have carefully analyzed the tradeoffs between three design factors that affect fusion performance: modality independence, curse of dimensionality, and fusion-model complexity. They demonstrated the utility of their scheme for image classification and video concept detection.

Feng et al. [28] proposed a bootstrapping framework for the annotation and retrieval of WWW images. In this work, the authors have adopted a co-training approach to annotate a large set of unlabeled samples using two orthogonal classifiers - one based on text, and the other on visual content features. In the co-training approach, two orthogonal classifiers independently confirm the quality of newly annotated samples based on their confidence level, and learn from each other's results.

Xu and Chua [90] proposed a layered framework to fuse the audio-visual features with the external knowledge such as match reports and game logs in order to detect events in team sports video. They first used hierarchical HMM for audio-visual event detection (feature-level fusion), and then combined the processed information from text sources with audio-visual information based on time-alignment (decision-level fusion). They adopted three different fusion schemes at decision level - Rule-based, Linear weighted sum (they call it 'agreegation'), and Bayesian inference. The authors provided a comparison of different fusion strategies and found that different fusion strategies are good under different conditions.

This thesis also adopts a hybrid approach in terms of level at which the assimilation takes place. We employ early (feature level) assimilation as well as late (decision level) assimilation strategy. We perform the feature-level assimilation only at the intra-media stream level and the decision-level assimilation approach at inter-media stream level. Since each media stream provides various features (such as blob's location and area in case of a video stream), their assimilation is performed locally for each media stream to obtain a local decision. Once all the local decisions are available, a global decision is derived by assimilating the local decisions incorporating their agreement and confidence information. *The late assimilation strategy has an advantage over early assimilation in that the former offers scalability (i.e. graceful upgradation or degradation) in terms of media streams used in the assimilation process* [5]. Note that, in late assimilation, we consider the media streams to be "decision-wise correlated". The decision-wise correlation refers to how the decisions obtained based on different media streams co-vary with each other. Our approach is different from Wu et al. [89] and Xu and Chua [90] in that we utilize the agreement/disagreement

information of streams instead of cross-correlation between the features of various heterogeneous streams. Feng et al. [28] also did not utilize the agreement/disagreement information.

2.1.5 Use of non audio-visual sensors for surveillance

There are few works which have demonstrated the use of sensors other than video and audio. Pavlidis and Faltesek [67] used bio-chemical sensors and video camera to propose a security system against bio-chemical attacks. In [68], Pavlidis et al. discussed about thermal near infrared solution for automatically counting the vehicle occupants. Cande et al. [16] proposed to use CMOS imagers for the detection and tracking of moving objects. Foresti and Snidaro [30] used infrared cameras and color cameras to build a distributed sensor network for video surveillance for outdoor environments. They employed a linear fusion for combining the trajectory information about objects. Peralta and Peralta [69] presented a Perimeter Intruder Detection System (PIDS) for surveillance of risky environments such as swimming pools. They used infrared sensor-emitter and detectors units driven by the micro-controllers. Recently, Prati et al. [71] also presented a multisensor surveillance system consisting of video cameras and passive infra-red sensors (PIR). Their proposed system helps in better object/person tracking.

However, all the works (except [30]) described above have not formally elaborated on the issue of fusion of data obtained from these heterogeneous sensors.

2.2 Use of Agreement/Disagreement Information

Our work is different from the works cited above in following aspects (Refer to Table 2.2). We explicitly compute and utilize the correlation informa-

tion (we call it the "agreement coefficient") among the streams at decision-level. The agreement coefficient among streams is computed based on how concurring or contradictory evidences they provide. Intuitively, higher the agreement among the streams, more would be the confidence in the global decision, and vice versa [73]. The various forms of correlation coefficients that have been used for diverse applications are based on content-wise dependency between the sources, hence are not suitable in our case. Pearson's correlation coefficient has been widely used as a measure of correlation among streams at feature-level, but like Lin's concordance correlation coefficient [53] and Kappa coefficient for the measure of agreement [13], it cannot be used in our case since it is evaluated to zero when the covariance among the observations is zero. In our case, if the decisions obtained on any two streams are similar, the agreement coefficient between the two should be high; however, using the existing measures of correlation, the covariance between the decisions obtained based on the two streams is evaluated to zero. Hence, these measures of correlation are unsuitable in our case. Therefore, the proposed framework models the agreement coefficient and its evolution based on the accumulated past history of how agreeing or disagreeing the media streams have been in their decisions [4]. That is how we use the past experience in our proposed information assimilation framework.

2.3 Use of Confidence Information

As shown in Table 2.2, most of the past works in multimodal fusion literature do not consider the notion of having confidences in the different modalities. We incorporate the stream's confidence information.

The confidence has also been used in the context of data management in sensor networks by Tatbul et al. [80]. Tavakoli et al. [81] also proposed

a method for event detection that uses historical and spatial information in clusters in order to determine a confidence level that warrants a detection report with high confidence. Similar to Tatbul et al. [80], we compute the confidence in a stream based on how it has helped in making the accurate decisions in the past. However, the works at [80] and [81] did not elaborate on how the confidence value is used in the integration of information. Moreover, we also propose a method to fuse the confidence in a group of streams.

In the context of multimedia, Feng et al. [28] utilized the confidence information for the different event detectors for annotating and retrieving WWW images. In this work, however, the authors have shown the use of confidence information only for the two detectors; while in our framework, it is generalized to any number of media streams. Also, the notion of fusion of confidences has not been used by Feng et al. [28].

To integrate confidence into the assimilation process, we use consensus theory. Consensus theory provides a notion of combining the single probability distributions based on their weights [11]. In our case, we essentially do the same by assigning weights to different media streams based on their confidence information. If we have more confidence in a media stream, a higher weight is given to it. Several consensus rules have been proposed, however the most commonly used consensus rules are - *linear opinion pool*(LOP) and *logarithmic opinion pool* (LOGP). In linear opinion pool, non-negative weights are associated with the sources to quantitatively express the "goodness" of each source. The *logarithmic opinion pool* treats data sources to be independent and is equivalent to the Bayesian combination if the weights are equal. We use *logarithmic opinion pool* since it satisfies the assumption of conditional (content-wise) independence among media streams which is

Table 2.2: Usage of agreement coefficient and confidence information

The work	Use of Agreement coefficient	Use of Confidence information	Fusion of Agreement coefficient	Fusion of Confidence information
The works listed in Table 2.1 except Feng et al. [28]	No	No	No	No
Feng et al. [28]	No	Yes	No	No
Tavakoli et al. [81]	No	Yes	No	No
Tatbul et al. [80]	No	Yes	No	No
Siegel and Wu [73]	No	Yes	No	Dempster-Shafer theory of evidence
Our approach	Yes	Yes	Yes	Bayesian formulation

essential to assimilation. The details are provided in Chapter 3.

Recently, Siegel and Wu [73] has also pointed out the importance of considering the confidence in sensor fusion. The authors have used the Dempster-Shafer (D-S) 'theory of evidence' to fuse the confidences. In contrast, we propose a model for confidence fusion by using a Bayesian formulation because it is both simple and computationally efficient [72].

2.4 Use of Contextual Information

The idea of 'context' has been primarily used in the areas of context-aware computing [88], knowledge-based systems [14, 82], and multimedia [85, 43, 77].

Wu [88] presented a context classification using human-centered approach, and decomposed context to the extent that it can be represented in a format of numerical values, string decompositions or indices.

Bremond and Thonnat [14] also provided a different classification primarily based on four types of information: scene environment, image acquisition, derived temporal, and user request. However, a formal definition of context is not provided by authors in [14] and [88].

Teriyan and Puuronen [82] introduced a formal model to represent con-

text. They used semantic meta network to represent context at multiple levels. Although their model is well formulated, it is more applicable to knowledge-based systems.

Jasinschi et al. [43] presented a layered probabilistic framework that integrates the multimedia content and context information. Within each layer the representation of content and context is based on Bayesian networks, and hierarchical priors provide the connection between the two layers. They applied the framework for an end-to-end system called Video Scout that selects, indexes, and stores TV program segments based on topic classification. Their work also does not formalize the context.

Sridharan et al. [77] introduced a formalization of 'context'. Their formal model to represent 'context' is also based on using semantic-nets. They define context as the union of semantic-nets, each of which can specify a fact about the environment. The inter-relationships amongst the various aspects (e.g. the user, the environment, the allowable interactions etc) of the system is used to define overall system context.

Similar to [85], we have used 'context' in terms of the environment information and the sensor information. The environment information consists of the geometry of the space under surveillance, the sensor information is related to their location and orientation. However, this thesis does not focus on the formalization of context.

2.5 Optimal Sensor Subset Selection

In the past, the optimal sensor selection problem has been widely studied in the context of discrete-event systems and failure diagnosis. The proposed approaches include an optimal measurement subsystem strategy [65], a Markovian decision strategy [26] and a formal method [46].

Table 2.3: A summary of approaches used for optimal sensor subset selection

The work	Confidence based selection	Cost based selection	Remarks
Oshman [65]	No	No	Static subset selection
Debouk et al. [26]	No	Yes	Uniform cost of all sensors unsuitable for heterogeneous sensor systems
Jiang et al. [46]	No	No	No tradeoff based on the cost of and the confidence in sensors
Lam et al. [51]	Accuracy based	No	Unsuitable for heterogeneous sensor systems
Pahalawatta et al. [66]	No	Energy	Main focus on energy consumption
Isler and Bajcsy [39]	No	No	No tradeoff based on the cost of and the confidence in sensors
Our approach	Yes	Yes	Dynamic programming based method which offers a tradeoff based on the cost of and the confidence in sensors

Oshman [65] proposed an optimal measurement subsystem strategy for discrete-time state estimators. At each sensor selection epoch, a measurement subsystem is selected, which contributes the largest amount of information along the principal state space direction. The method has a limitation that the *a priori* information about the sensors in a subsystem must be known. We overcome this limitation by dynamically forming such subsystems (we shall refer to them as 'subsets') during the execution of the algorithm. Moreover, this method does not consider the cost of the subsystems and the confidence, as incorporated in our framework.

Debouk et al. [26] formulated the optimization problem as a Markovian decision problem (MDP) with the objective to identify instances where it is possible to explicitly determine optimal strategies. The sequence of tests is applied to identify the least costly sensor combination that satisfies a set of system properties (such as diagnosability) with the minimum expected number of tests. The method works under the specified assumptions which are over-constrained. For instance, the authors assume an uniform cost for all sensors which is impractical in a multimedia environment where differ-

ent *types* of media are employed. This work also does not integrate the confidence in sensors, which our proposed framework does.

Jiang et al. [46] presented a formal method for optimal sensor selection for discrete event systems with partial observation. The sensor subset (or observation mask) that qualifies for selection must follow the desired formal properties such as (co-)observability, or normality (for control), the state-observability (for state-estimation), and the diagnosability (for failure diagnosis) under partial observation etc. However, this method does not consider the cost of obtaining a subset of sensors, and the system designer's confidence in this subset while attempting to locate the optimal observation mask.

A sensor selection method for the execution of continuous probabilistic queries has also been proposed by Lam et al. [51]. This method meets the accuracy requirement by selecting the set of sensors which are highly correlated. The correlation is computed assuming that all the sensors are of same type. Therefore, their method is not suitable for a set of heterogeneous sensors. Also, they do not explicitly consider the cost of each sensor.

In the context of wireless sensor networks, Pahalawatta et al. [66] proposed to solve the problem of optimal sensor selection by maximizing the information utility gained from a set of sensors subject to a constraint on the average energy consumption in the network. However, their method does not consider the confidence in sensors. Moreover, our framework also takes into account of the processing cost of sensor data.

Recently, Isler and Bajcsy [39] proposed a generic sensor model where the measurements can be interpreted as polygonal, convex subsets of the plane. They used an approximation algorithm so as to minimize the error in estimating the position of a target. However, this work also does not

explicitly have a notion of the cost of using streams and the confidences in them.

A summary of the methods for optimal subset sensor selection is presented in Table 2.3. In contrast to all the solutions described above, our proposed work is different in that our framework provides a *tradeoff* between the extent to which the goal is achieved, the confidence in the streams and the cost of using streams. In addition, our method also provide a flexibility to the system designer to choose next best sensor if the best sensor is not available.

Chapter 3

Information Assimilation

In this chapter, we describe the proposed framework for information assimilation and focus on two issues - 'when' and 'how' to assimilate the information obtained from different sources. The details of how the framework addresses the third issue (i.e. 'what' to assimilate) will be described in Chapter 4. This chapter begins with the problem formulation in section 3.1. In section 3.2, we provide a overview of the proposed information assimilation framework for event detection in multimedia surveillance and monitoring systems. Section 3.3 elaborates on the issue of timeline-based event detection. We describe the hierarchical probabilistic method used for information assimilation in section 3.4. Finally, in section 3.5, we present simulation results to demonstrate the utility of considering agreement coefficient in the assimilation process.

3.1 Problem Formulation

We use the following model of computation:

$\mathcal{M}1$ **S** is a multimedia surveillance and monitoring system designed for de-

tecting a set E of N_E number of events, and it consists of n heterogeneous sensors that capture data from the environment. Let $\mathbf{M}^n = \{M_1, M_2, \ldots, M_n\}$ be the media streams obtained from n sensors.

$\mathcal{M}2$ For $1 \leq i \leq n$, let t_i be the *minimum time interval* in which the decision about an event are obtained based on stream M_i. This minimum time interval includes the amount of time in which the data is captured from the sensor device and in which it is processed.

$\mathcal{M}3$ For $1 \leq i \leq n$, let $0 < p_i < 1$ be the *probability* of occurrence of an event based on individual i^{th} media stream. The p_i is determined by first extracting the features from media stream i and then by employing an event detector (e.g. a trained classifier) on them. Also, let P_Φ be the 'fused probability' of occurrence of the event based on a subset $\Phi \in \mathcal{P}(\mathbf{M}^n)$ of media streams. The 'fused probability' is the overall probability of occurrence of the event based on a group of media streams [1].

$\mathcal{M}4$ For $1 \leq i \leq n$, c_i, let be the *cost* per unit time of using stream i. Also, $C_n = \sum_{i=1}^{n} c_i$ be the *total cost*. The cost of a stream usually includes the installation cost of sensor device, its operating cost and the processing cost of stream. In our case, we determine the cost of streams based on their processing time.

$\mathcal{M}5$ For $1 \leq i \leq n$, let $0.5 < f_i < 1$ be the system designer's *confidence* in the i^{th} stream. The confidence in a stream is learned by experimentally determining its accuracy. More the accurate results we obtain based on a stream, more the confidence we would have in it.

We make the following assumptions:

$\mathcal{A}1$ All sensing devices capture the same environment (but optionally, the different aspects of the environment) and provide correlated observations.

$\mathcal{A}2$ The system designer's confidence level in each of the media streams is at least 0.5. This assumption is reasonable since it is not useful to employ a media device which is found to be inaccurate more than half of the time.

$\mathcal{A}3$ The fused probability of the occurrence of event and the overall confidence increase monotonically as the more concurring evidences are obtained from the streams.

$\mathcal{A}4$ Though the minimum detection time interval could be different for different events when detected based on different streams, we assume it to be the same for all the events. Relaxing this assumption is an open problem which is out of the scope of this thesis and will be explored in future work.

$\mathcal{A}5$ The system can detect multiple events and each event can be detected by using a subset of total number of streams. Hence, there is a need to select the best subset for a specific event.

The objective is to determine:

1. The overall probability P_Φ of the occurrence of event based on subset $\Phi \in \mathcal{P}(\mathbf{M}^n)$ of streams.

2. The time interval t_w at which the overall probability should be computed.

3. The optimal subset $\Phi \in \mathcal{P}(\mathbf{M}^n)$ of streams under the specified constraints. We formulate three different problems referred to as the *Multimedia Selection* (MS) Problems **MaxGoal**, **MaxConf** to **MinCost** as follows:

   Find the subset $\Phi \in \mathcal{P}(\mathbf{M}^n)$ that -

   Problem MaxGoal : maximizes P_Φ
   subject to $C_\Phi \leq C_{spec}$ and $F_\Phi \geq F_{spec}$.

   Problem MaxConf : maximizes F_Φ
   subject to $C_\Phi \leq C_{spec}$ and $P_\Phi \geq P_{spec}$.

   Problem MinCost : minimizes C_Φ
   subject to $F_\Phi \geq F_{spec}$ and $P_\Phi \geq P_{spec}$.

   The notations used are:

   P_Φ is the fused probability of the occurrence of event when the subset Φ of media streams is used by system $\mathbf{S}$.

   C_Φ is the cost of using the subset Φ of streams.

   F_Φ is the overall confidence when the subset Φ of streams is used.

   P_{spec} is the specified minimum fused probability of the occurrence of event.

   C_{spec} is the specified maximum overall cost. Note that $C_\Phi \leq C_n$.

   F_{spec} is the specified minimum overall confidence.

In this chapter, we focus on the solutions of issues (1) and (2) mentioned above. The solution of the issue (3) will be discussed in Chapter 4.

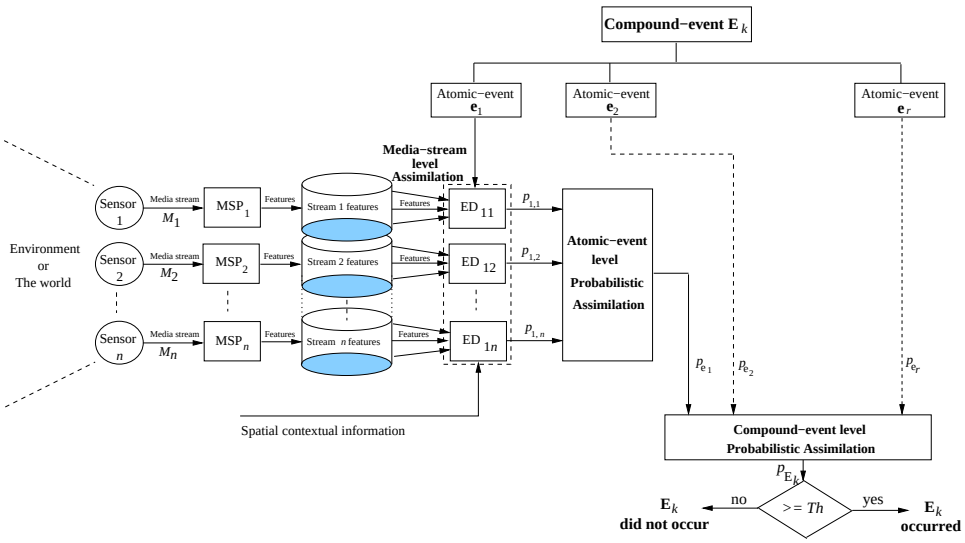

Figure 3.1: A schematic overview of the hierarchical approach used in information assimilation framework for the detection of an event $\mathbf{E}_k$ in a surveillance system consisting of n sensors

3.2 Overview of the Framework

The proposed information assimilation framework [4] adopts a hierarchical probabilistic approach in order to detect an event in a surveillance and monitoring environment, and performs assimilation of information at three different hierarchical levels - media stream level, atomic event level and the compound event level. The work flow of the framework is depicted in figure 3.1. The media streams obtained from n sensors are processed using respective *Media Stream Processors* (MSP_1 to MSP_n). Each MSP_i, $1 \leq i \leq n$, is a set of media processing tools that extracts features from the media stream M_i; for example, a blob detector extracts blobs from a video stream. The features extracted from each media stream are stored in their respective databases.

Let the system detect N_a number of atomic events (given by $\mathbf{e}_1$, $\mathbf{e}_2$, ..., $\mathbf{e}_{N_a}$). The total number of sets containing two or more atomic events in

which the atomic events can occur together is given by $\sum_{r=2}^{N_a} \binom{N_a}{r}$. Any k^{th} compound event $\mathbf{E}_k$ can be expressed as $\mathbf{E}_k = \langle \mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_r \rangle$, where $2 \leq r \leq N_a$, $1 \leq k \leq N_c$, N_c being the number of compound events which can be detected by using the system. The total number N_E of events (atomic events as well as compound events) can be given by $N_E = N_a + N_c$.

A compound event $\mathbf{E}_k$, which comprises of two or more atomic events occurring together, is detected hierarchically in a bottom-up manner as shown in figure 3.1. First, atomic events $\mathbf{e}_j$, $1 \leq j \leq r$ are detected using the relevant media streams, and then these decisions are assimilated hierarchically to obtain an overall decision for the compound event $\mathbf{E}_k$, as will shortly be described in section 3.4.

From the total number N_a of atomic events that the system can detect, the proposed framework identifies -

- The atomic events (e.g. person's standing/walking/ running and person's talking/shouting) that cannot occur simultaneously.
- The atomic events (e.g. person's walking) that can occur individually as well as can occur together with some other atomic event (e.g. with person's shouting).
- The atomic events (such as person's shouting) that cannot occur individually and must occur together with some other atomic event (such as with person's standing/walking/running).

Next, the framework also identifies the types of streams based on which these atomic events can be detected. Note that, for the identification of atomic and compound events, and also for the identification of which atomic event could be detected based which stream, the domain knowledge is provided externally.

Table 3.1: All possible events in Example 3.1

Event number	Constituent atomic events
1	Standing
2	Walking
3	Running
4	Standing , Talking
5	Standing, Shouting
6	Standing, Door knocking
7	Walking, Talking
8	Running, Talking
9	Walking, Shouting
10	Running, Shouting
11	Standing, Talking, Door knocking
12	Standing, Shouting, Door knocking

To further illustrate it, we provide the following example.

Example 3.1 Let us consider a surveillance system that uses two types of sensors - video and audio with the goal of detecting $N_a = 6$ atomic events, namely - person's "standing", "walking", "running", "talking", "shouting" and "door knocking". In this case, as shown in Table 3.1, there could be $N_c = 9$ compound events in which any $r \geq 2$ atomic event(s) could occur. In total, there could be $N_E = 12$ events. The atomic events in this example can be detected as follows - standing (V), walking (AV), running (AV), talking (A), shouting (A), door knocking (A); where (A), (V) and (AV) denote audio, video and audio-video streams, respectively.

3.3 Timeline-based Event Detection

As discussed earlier in section 1.1 of Chapter 1, the events occur over a timeline. There are various issues related to timeline-based event detection such as -

- To mark the start and end of an event over a timeline, there is a need to obtain and process the data streams at certain time intervals [75]. This time interval, which is basically the minimum amount of time

to confirm an event, could be different for different atomic/compound events when detected using different data streams. Determining the minimum time period to confirm different events is a research issue which is out of scope of this thesis and will be explored in the future work. In this dissertation, we assume this minimum time period to be the same for all the atomic/compound events (Refer to assumption $\mathcal{A}4$ in section 3.1).

- Determining the minimum time period for a specific atomic event is also critical. Ideally, it should be as small as possible since a smaller value of it allows to detect the events at a finer granularity in time. The minimum time period for a specific atomic event should be just large enough to capture the data to confirm it. We learn its suitable value through experiments.

- Since the information from different sources become available at different time instances, when should it be assimilated is another research issue. There could be several strategies to resolve this issue. We assimilate the information at fixed time intervals t_w. This time interval is determined as -

$$t_w = \max_{i=1:n} (t_i) \tag{3.1}$$

i.e. by choosing the maximum of all the minimum time periods in which various atomic events can be confirmed. Although this strategy may not be the best, it is computationally less-expensive. Again, exploring other strategies is an issue which will be considered in the future.

3.4 Hierarchical Probabilistic Assimilation

The proposed framework adopts a hierarchical probabilistic assimilation approach and performs assimilation of information obtained from diverse data sources at three different levels - Media stream level, Atomic event level and Compound event level. The details are as follows.

3.4.1 Media stream level assimilation

As shown in figure 3.1, the *Event Detectors* (ED_{ji}, $1 \leq j \leq r$ and $1 \leq i \leq n$) are employed to independently detect each atomic event $\mathbf{e}_j$ based on the respective features obtained from media streams M_i, $1 \leq i \leq n$. At media stream level, all the available features from a media stream are combined. The event detectors make the decision about an atomic event based on the combined features. Whenever required, they also utilize the contextual information (environment information, in our case) such as the geometry of the monitored space, location, orientation and the coverage space etc of sensors. The event detectors provide their decisions in probabilities $p_{j,i}$, $1 \leq j \leq r$ and $1 \leq i \leq n$ (Figure 3.1). The $p_{j,i}$ implies probability of the occurrence of atomic event $\mathbf{e}_j$ based on media stream M_i.

3.4.2 Atomic event level assimilation

At the next level, since the decisions about an atomic event $\mathbf{e}_j$, that are obtained based on all the relevant media streams, may be similar or contradictory; these decisions are assimilated using a Bayesian approach incorporating streams' agreement/disagreement and confidence information. For the atomic events $\mathbf{e}_j$, $1 \leq j \leq r$, the framework follows the steps -

1. At any particular instant, all the streams are grouped into two subsets

S_1 and S_2. S_1 and S_2 contain the streams based on which the event detectors provide decision in favor and against the occurrence of the atomic event, respectively. Precisely, the streams based on which the system estimates the probability of the occurrence of event more than 0.50 are put in set S_1 and the rest in set S_2.

2. We also experimentally learn the confidence level f_i of each stream M_i, $1 \leq i \leq n$ by letting the system used only the stream M_i for detecting an event. The confidence level is assigned to a stream based on how it has helped in accurately detecting an event.

3. Using the streams in the two subsets S_1 and S_2, we compute overall probabilities $P(\mathbf{e}_j|S_1)$ and $P(\bar{\mathbf{e}_j}|S_2)$ of occurrence and non-occurrence of the atomic event $\mathbf{e}_j$, respectively. The overall probabilities are computed using a Bayesian assimilation approach which will be described shortly. We also find the overall confidence F_{S_1} and F_{S_2} for the subsets S_1 and S_2, respectively. The method of finding the overall confidence in a group of streams will be described in section 3.4.2.

4. The weights to two subsets are assigned based on their respective overall confidence values. If $P(\mathbf{e}_j|S_1).F_{S_1} \geq P(\bar{\mathbf{e}_j}|S_2).F_{S_2}$, it is concluded that the atomic event $\mathbf{e}_j$ has occurred with a probability $p_{\mathbf{e}_j} = P(\mathbf{e}_j|S_1)$, else it did not occur with a probability $p_{\mathbf{e}_j} = P(\bar{\mathbf{e}_j}|S_2)$.

We assume the media streams to be "content-wise" independent. This assumption is reasonable since media streams may be of different types, and may have different data formats and representations. However, since the decision about the same atomic event is obtained based on all the streams, we can assume them to be "decision-wise" correlated.

We describe in the following paragraphs how the assimilation of decision-wise correlated media streams takes place, and also how the agreement coefficient and confidence information about them are modeled.

Assimilation of correlated media streams

As shown in figure 3.1, the system outputs local decisions $p_{j,i}$ (also denoted as $P(\mathbf{e}_j|M_i)$), $1 \leq i \leq n$, $1 \leq j \leq r$, about an atomic event $\mathbf{e}_j$. Along a timeline, as these probabilistic decisions are available, we iteratively integrate all the media streams using a Bayesian approach. The proposed approach allows for incremental and iterative addition of new stream. Let $P(\mathbf{e}_{j_t}|\mathbf{M}_t^{i-1})$ denote probability of the occurrence of atomic event $\mathbf{e}_j$ at time t based on media streams M_1, M_2, ..., M_{i-1}. The updated probability $P(\mathbf{e}_{j_t}|\mathbf{M}_t^{i})$ (i.e. the overall probability after assimilating the new stream $M_{i,t}$ at time instant t) can be iteratively computed as:

$$P(\mathbf{e}_{j_t}|\mathbf{M}_t^{i}) = \frac{P(M_{i,t}|\mathbf{e}_{j_t})P(\mathbf{e}_{j_t}|\mathbf{M}_t^{i-1})}{P(M_{i,t}|\mathbf{M}_t^{i-1})} \tag{3.2}$$

In the above equation, the term $P(M_{i,t}|\mathbf{e}_{j_t})$ denotes the likelihood of occurrence of atomic event $\mathbf{e}_{j_t}$ based on i^{th} stream $M_{i,t}$ at time t [1]. The term $P(\mathbf{e}_{j_t}|\mathbf{M}_t^{i-1})$ is posterior probability of occurrence of atomic event $\mathbf{e}_{j_t}$ based on $i-1$ streams and this term becomes prior when i^{th} stream is integrated. The term $P(M_{i,t}|\mathbf{M}_t^{i-1})$ serves as a normalization function to ensure that the posterior probabilities sum to one over the occurrence and non-occurrence of the atomic event $\mathbf{e}_{j_t}$. The equation (3.2) can be re-written as follows:

$$P(\mathbf{e}_{j_t}|\mathbf{M}_t^{i}) = \alpha_i P(\mathbf{e}_{j_t}|\mathbf{M}_t^{i-1})P(M_{i,t}|\mathbf{e}_{j_t}) \tag{3.3}$$

where, α_i is a normalization factor.

Equation (3.3) shows the assimilation using Bayesian approach under the assumption that all the media streams have equal confidence levels and zero agreement coefficient. In what follows, we relax this assumption and integrate the agreement/disagreement and confidence information of media streams in their assimilation.

The confidence in each media stream is computed by experimentally determining its accuracy. To integrate the confidence into assimilation process, we use consensus theory. Consensus theory provides a notion of combining the single probability distributions based on their weights [11]. In our case, we essentially do the same by assigning weights to different media streams based on their confidence information. If we have more confidence in a media stream, a higher weight is given to it. Several consensus rules have been proposed, however the most commonly used consensus rules are - *linear opinion pool* (LOP) and *logarithmic opinion pool* (LOGP). In linear opinion pool, non-negative weights are associated with the sources to quantitatively express the "goodness" of each source. The rule is formulated as -

$$T_c(p_1, p_2, \ldots, p_n) = \sum_{i=1}^{n} w_i' p_i \tag{3.4}$$

where, $p_i, 1 \leq i \leq n$, are the individual probabilistic decisions; and w_i', $1 \leq i \leq n$ are their corresponding weights whose sum is equal to 1 i.e. $\sum_{i=1}^{n} w_i' = 1$.

In *logarithmic opinion pool*, the data sources are treated to be independent, and it formulation is similar to Bayesian formulation when the weights are equal. Similar to linear opinion pool, in logarithmic opinion pool strategy also, the weights are non-negative and they represent the sensors "goodness". Since we adopt a Bayesian inference model in the assimilation process

which assumes the (content-wise) independence among media streams, the *logarithmic opinion pool* becomes more suitable for our assimilation model.

The logarithmic opinion pool rule is described as [32] -

$$log[T_c(p_1, p_2, \ldots, p_n)] = \sum_{i=1}^{n} w_i' log(p_i) \tag{3.5}$$

or

$$T_c(p_1, p_2, \ldots, p_n) = \prod_{i=1}^{n} {p_i}^{w_i'} \tag{3.6}$$

where, $p_i, 1 \leq i \leq n$, are the individual probabilistic decisions and $\sum_{i=1}^{n} w_i' = 1$. We normalize it over the two aspects of an event - the occurrence and non-occurrence of event. The formulation is shown as -

$$T_c(p_1, p_2, \ldots, p_n) = \frac{\prod_{i=1}^{n} {p_i}^{w_i'}}{\sum_{E_k}(\prod_{i=1}^{n} {p_i}^{w_i'})} \tag{3.7}$$

We use this formulation to develop the assimilation model which will be described shortly.

The agreement coefficient between two media streams is used as a scaling factor for the overall probability of occurrence of an event. The idea is that higher the agreement coefficient between the two media streams, higher would be the overall probability. We use this notion in the proposed assimilation model.

The assimilation model that combines the probabilistic decisions based on two sources $\mathbf{M}^{i-1}$ (i.e. a group of $i-1$ streams) and M_i (i.e. an individual i^{th} stream) is given as follows:

$$P_{j,i} = \frac{(P_{j,i-1})^{F_{i-1}}.(p_{j,i})^{f_i}.e^{\overline{\gamma}_i}}{(P_{j,i-1})^{F_{j,i-1}}.(p_{j,i})^{f_i}.e^{\overline{\gamma}_i} + (1 - P_{j,i-1})^{F_{i-1}}(1 - p_{j,i})^{f_i}.e^{-\overline{\gamma}_i}} \tag{3.8}$$

where, $P_{j,i} = P(\mathbf{e}_{j_t}|\mathbf{M}_t^i)$ and $P_{j,i-1} = P(\mathbf{e}_{j_t}|\mathbf{M}_t^{i-1})$ are the probabilities of

occurrence of atomic event $\mathbf{e}_j$ using $\mathbf{M}^i$ and $\mathbf{M}^{i-1}$, respectively, at time instant t. $p_{j,i} = P(\mathbf{e}_{j_t}|M_{i,t})$ is probability of the occurrence of atomic event $\mathbf{e}_j$ based on only i^{th} stream at time instant t. Similarly, F_{i-1} and f_i (such that $F_{i-1}+f_i = 1$) are the confidence in $\mathbf{M}^{i-1}$ and M_i, respectively. The computation of confidence for a group of media streams will be described shortly. The $\overline{\gamma}_i \in [-1, 1]$ is the agreement coefficient between two sources $\mathbf{M}^{i-1}$ and M_i. The limits -1 and 1 represent full disagreement and full agreement, respectively, between the two sources. The modeling of $\overline{\gamma}_i$ is described in subsequent paragraphs.

Modeling of the agreement coefficient

The correlation among the media streams refers to the measure of their agreement or disagreement with each other. We call this measure of agreement to be the "Agreement Coefficient" among the streams [3]. Let the measure of agreement among the media streams at time t be represented by a set $\Gamma(t)$ which is expressed as:

$$\Gamma(t) = \{\gamma_{ii'}(t)\} \tag{3.9}$$

where, $1 \leq i, i' \leq n$ and the term $-1 \leq \gamma_{ii'}(t) \leq 1$ is the *agreement coefficient* between the media streams M_i and $M_{i'}$ at time instant t.

The *agreement coefficient* $\gamma_{ii'}(t)$ between the media streams M_i and $M_{i'}$ at time instant t is computed by iteratively averaging the past agreement coefficients with the current observation. The $\gamma_{ii'}(t)$ is precisely computed as:

$$\gamma_{ii'}(t) = \frac{1}{2}\left[(1 - 2 \times abs(p_i(t) - p_{i'}(t))) + \gamma_{ii'}(t-1)\right] \tag{3.10}$$

where, $p_i(t) = P(\mathbf{e}_{j_t}|M_i)$ and $p_{i'}(t) = P(\mathbf{e}_{j_t}|M_{i'})$ are the individual probabilities of occurrence of atomic event $\mathbf{e}_j$ based on media streams M_i and

$M_{i'}$, respectively, at time $t \geq 1$; and $\gamma_{ii'}(0) = 1 - 2 \times abs(p_i(0) - p_{i'}(0))$. In equation (3.10), the term $(1 - 2 \times abs(p_i(t) - p_{i'}(t))$ denotes the agreement/disagreement at the current instant t and the term $\gamma_{ii'}(t-1)$ denotes the accumulated past agreement coefficient between the streams M_i and $M_{i'}$. These probabilities represent decisions about the atomic events. Exactly same probabilities would imply full agreement ($\gamma_{ii'} = 1$) whereas totally dissimilar probabilities would mean that the two streams fully contradict each other ($\gamma_{ii'} = -1$). Note that any three media streams, in agreeing/disagreeing with each other, do follow the commutativity rule.

The agreement coefficient between two sources $\mathbf{M}^{i-1}$ and M_i is modeled as:

$$\overline{\gamma}_i = \frac{1}{i-1} \sum_{s=1}^{i-1} \gamma_{si} \tag{3.11}$$

where, γ_{si} for $1 \leq s \leq i-1$, $1 < i \leq n$ is the agreement coefficients between the s^{th} and i^{th} media streams. The agreement fusion model given in equation (3.11) is based on *average-link clustering*. In average-link clustering, we consider the distance between one cluster and another cluster to be equal to the average distance from any member of one cluster to any member of the other cluster. In our case, a group $\mathbf{M}^{i-1}$ of $i-1$ media streams is one cluster and we find the average distance of new i^{th} media stream with this cluster. Note that the fused agreement coefficient $\overline{\gamma}_i$ is used for combining M_i with $\mathbf{M}^{i-1}$ as described before in equation (3.8).

Confidence fusion

In the context of streams, the confidence in a stream is related to its accuracy. The higher the accuracy of a stream, higher the confidence we would have in it. We compute the accuracy of a stream by determining how many times an event is correctly detected based on it out of the total number of

tries. Note that, in our case, the accuracy of a stream includes the measurement accuracy of the sensor as well as the accuracy of the algorithm used for processing the stream.

The *confidence fusion* refers to the process of finding the overall confidence in a group of media streams where the individual media streams have their own confidence level. If the two streams M_i and $M_{i'}$ have their confidence levels f_i and $f_{i'}$, respectively; what would our confidence be in a group which contains both the streams? The intuitive answer to this question would be that our overall confidence should increase as the number of streams increases. Considering the confidence values as the probabilities, we propose a Bayesian method to fuse the confidence levels in individual streams. The overall confidence $f_{ii'}$ in a group of two media streams M_i and $M_{i'}$ is computed as follows:

$$f_{ii'} = \frac{f_i \times f_{i'}}{f_i \times f_{i'} + (1 - f_i) \times (1 - f_{i'})} \tag{3.12}$$

In the above formulation, we make two assumptions. First, we assume that the system designer's confidence level in each of the media streams is more than 0.5. This assumption is reasonable since there is no use of employing a sensor which is found to be inaccurate more than half of the time. Second, although the media streams are correlated in their decisions; we assume that they are mutually independent in terms of their confidence levels.

For n number of media streams, the overall confidence is iteratively computed. Let F_{i-1} be the overall confidence in a group of $i-1$ streams. By fusing the confidence f_i of i^{th} stream with F_{i-1}, the overall confidence F_i in

a group of i streams is computed as:

$$F_i = \frac{F_{i-1} \times f_i}{F_{i-1} \times f_i + (1 - F_{i-1}) \times (1 - f_i)} \tag{3.13}$$

3.4.3 Compound event level assimilation

At the compound event level, the overall probability $p_{\mathbf{E}_k}$ of the occurrence of compound event $\mathbf{E}_k$ is estimated by assimilating the probabilistic decisions p_{e_j}, $1 \leq j \leq r$ about the r atomic events by using the following assimilation model -

$$p_{\mathbf{E}_k} = \frac{\prod_{j=1}^{r} p_{e_j}}{\prod_{j=1}^{r} p_{e_j} + \prod_{j=1}^{r} (1 - p_{e_j})} \tag{3.14}$$

If $p_{\mathbf{E}_k}$ is found greater than the threshold Th, the system decides in favor of the occurrence of compound event $\mathbf{E}_k$, else it decides against it.

Since the atomic events are independent, the agreement coefficients among them are considered as zero, and hence is not integrated into equation (3.14). For example, atomic events $\mathbf{e}_1$ = "A person is walking in the corridor" and $\mathbf{e}_2$ = "A person is shouting in the corridor" are essentially independent since a person's walking is completely independent of the person's shouting. The confidence information is also not integrated into this assimilation model because the confidence is usually associated with media streams and not with the atomic events.

3.5 Simulation Results

In this section, we present simulation results in order to show how agreement coefficient between streams plays an important role in improving the overall (fused) probability of detecting the event. Note that the experimental results in greater details in a real surveillance setup will be provided in Chapter 5,

section 5.2.

The synthetic data for simulation consists of 100 media streams, based on each of which we are able to detect an arbitrary event with an uniform probability. The simulation of the assimilation process is performed with an objective to study the affect of agreement coefficient on overall fused probability [1].

In figure 3.2, we show only up to 15 streams since after the assimilation of 15 streams the fused probability is close to maximum in both cases (figure3.2a-3.2b). To show how only agreement coefficient can affect the assimilation, we assume that all the media streams are equi-probable of helping detecting the event, and we also assume that there is uniform agreement coefficient among all the streams. The simulation is performed for two types of stream sets. The streams within each set have uniform probabilities which are 0.60 and 0.80 (figure 3.2a to 3.2b, respectively). For each set of streams, these probabilities are assimilated sequentially (using equation (3.8)) with the agreement coefficients 0.0, +0.5, +1.0.

We did not consider negative agreement coefficient for sake of simplicity. Considering negative agreement coefficient would require streams to follow the commutativity rule in agreeing or disagreeing with each other (as discussed in section 3.4.2). Also, we restricted this simulation to study only agreement coefficient and did not consider the confidence information. However, the effect of considering confidence and agreement/disagreement information on real data will be shown in Chapter 5.

Our observations from the graphs (in figure 3.2) are -

- From figure 3.2a, we observed that the system can attain the fused probability close to maximum based on a few streams (lesser than 5) with high agreement coefficients (+0.5 and +1.0). It is also observed

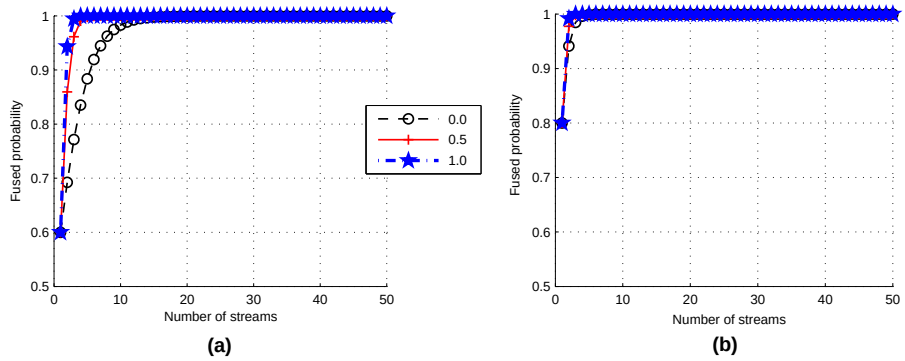

Figure 3.2: Fused probability vs. Number of media streams (with uniform probabilities (a) 0.60 (b) 0.80, for all streams)

that, with zero agreement coefficient, around 15 streams having moderate probabilities can still help in detecting the event.

- As shown in figure 3.2b, if the streams having high individual probabilities and high agreement coefficient are assimilated, even very few streams can help in detecting the event. E.g. two streams with probabilities 0.80 and agreement coefficient +1.0 are adequate in helping detecting the event. Note that, more the number of streams, higher would be the time taken to assimilate them and to make a decision.

- These results suggest that streams having higher individual probabilities is better, but agreement coefficient also plays an important role in improving the overall probability. This indicates that a few but the streams having high agreement coefficient can better help in detecting the event.

Chapter 4

Optimal Subset Selection of Media Streams

In this chapter, we describe how the proposed framework addresses the issue of 'what to assimilate' in order to accomplish a surveillance task. The framework uses a dynamic programming based method for finding the optimal subset of streams in order to detect events in surveillance and monitoring systems. Note that, in the previous chapter, we elaborated on 'when to assimilate' and 'how to assimilate' issues of the framework.

In section 3.1 of Chapter 3, we formulated three different MS problems - **MaxGoal**, **MaxConf** and **MinCost** for finding the optimal subset of streams under the specific constraints. In this chapter, we describe in detail these problems with a focus on proving them to be NP-Complete problems and also on providing dynamic programming based solutions to them for finding the optimal subsets in *pseudo*-polynomial time.

This chapter is organized as follows. We first provide an introduction to these problems in section 4.1. We then discuss the computational complexities of these problems and prove them to be NP-Complete in section 4.2.

Next, in section 4.3, we discuss the basis for developing solutions to three MS problems. In section 4.4, we present the proposed dynamic programming based methods to solve these three problems **MaxGoal**, **MaxConf** and **MinCost**. We discuss the time and space complexities of the proposed algorithms with a comparison to the brute-force approach in section 4.5. Finally, in section 4.6, we supplement with simulation results to show the utility of the proposed dynamic programming based method.

4.1 Introduction

To accomplish a task, which subset of media streams is the optimal one? This question can be answered in many ways. The optimal subset may be the one which maximizes the probability of achieving the system goal subject to a certain level of confidence or the specified cost. The system goal in our case is the detection of an event. Higher the probability of the occurrence/non-occurrence of event we obtain, more is the chances that the system goal is accomplished. The optimal subset may also be the one which minimizes the cost subject to the specified extent to which the goal is achieved with a certain level of confidence. The subset which maximizes the overall confidence under a specified cost can also be considered as the optimal subset - which is what we intend to determine. We thus study the problem of optimal stream selection from the following three different angles:

1. Maximizing the probability of the occurrence/non-occurrence of event under the specified maximum cost and with a specified minimum confidence.

2. Maximizing the confidence in the media streams used with a specified

minimum probability of the occurrence/non-occurrence of event under a specified maximum cost.

3. Minimizing the cost of using the media streams to attain a specified minimum probability of the occurrence/non-occurrence of event with a specified minimum confidence.

We reduce the 0-1 KNAPSACK problem [50] to the problem of optimal media selection and use a dynamic programming approach to solve it [2, 5]. In our problem, for each media stream, the probability of the occurrence/non-occurrence of event based on it and the system designer's confidence level in it are analogous to the *profit*, while its cost is analogous to the *weight* of a KNAPSACK problem. The fundamental difference is that we fuse the probabilities and confidence levels using a Bayesian approach [1], while the profits are simply added in the 0-1 KNAPSACK problem.

From a theoretical perspective, the problem is proven to be NP-Complete. Thereafter, the proposed framework uses a dynamic programming approach that finds the optimal subset of streams based on the above three criteria. From an AI point of view, the solution we propose is heuristic-based, and for each criterion, it utilizes a heuristic function which, for a given problem, combines *optimal* solutions of small-sized sub-problems to yield a potential near-optimal solution to the original problem. To achieve the latter, we resort to a recent result proven in [64], where Oommen and Rueda showed that the quality of a heuristic algorithm is determined by the accuracy of the heuristic function it uses.

4.2 Complexity of Computing Optimal Solutions to the MS Problems

In this section, we prove using Theorem 4.2.1 that the MS Problems are *NP-Complete* problems.

Theorem 4.2.1 *The MS Problems are NP-Complete problems, whenever the number of media streams* $n \geq 2$.

Proof: The three MS problems are the optimization problems. They can be restated as decision problems in the following manner -
MaxGoal = {Does a subset Φ, based on which we obtain a fused probability $P_\Phi \geq P_{spec}$ of the occurrence/non-occurrence of event, exist subject to the overall confidence in it is atleast F_{spec} and the overall cost of using it is atmost C_{spec}}
MaxConf = {Does a subset Φ, in which we have the overall confidence $F_\Phi \geq F_{spec}$, exist subject to the fused probability of the occurrence/non-occurrence of event based on it is atleast P_{spec} and the overall cost of using it is atmost C_{spec}}
MinCost = {Does a subset Φ, with overall cost $C_\Phi \geq C_{spec}$, exist subject to the fused probability of the occurrence/non-occurrence of event based on it is atleast P_{spec} and the overall confidence in it is atleast F_{spec}}

The proof for this theorem is similar for all the three problems, **MaxGoal**, **MaxConf** and **MinCost**. We consider the case of Problem **MaxGoal**. To prove Problem **MaxGoal** to be NP-Complete problem, we provide Lemmas 4.2.2, 4.2.3 and 4.2.4 which together prove Theorem 4.2.1.

Lemma 4.2.2 *The* 0-1 KNAPSACK *problem is reducible to problem* **MaxGoal** *in polynomial time* i.e. 0-1 KNAPSACK $\geq_{Polynomial}$ **MaxGoal**.

Proof: We pick a known NP-Complete 0-1 KNAPSACK problem and define an instance of it as a 5-tuple

$$\langle \mathbf{U}^n, \mathbf{X}, \mathbf{W}, X_{spec}, W_{spec} \rangle$$

with a set $\mathbf{U}^n = \{u_i\}_{i=1}^n$ of n items, their profits $\mathbf{X} = \{x_i\}_{i=1}^n$, weights $\mathbf{W} = \{w_i\}_{i=1}^n$, specified minimum profit X_{spec}, knapsack capacity W_{spec}; and with an objective of determining whether a subset $\Lambda \subseteq \mathbf{U}^n$ of items having overall profit $X_\Lambda \geq X_{spec}$ exists under the constraint $W_\Lambda \leq W_{spec}$, where W_Λ is the total weight of items of subset Λ.

The corresponding instance of **MaxGoal** is defined by a 7-tuple

$$\langle \mathbf{M}^n, \mathbf{P}, \mathbf{F}, \mathbf{C}, P_{spec}, C_{spec}, F_{spec} \rangle$$

with a set $\mathbf{M}^n = \{M_i\}_{i=1}^n$ of n streams, the probabilities $\mathbf{P} = \{p_i\}_{i=1}^n$ of the occurrence/non-occurrence of event based on individual streams, their confidences $\mathbf{F} = \{f_i\}_{i=1}^n$, costs $\mathbf{C} = \{c_i\}_{i=1}^n$, minimum specified fused probability P_{spec}, maximum specified cost C_{spec} and minimum specified confidence F_{spec}; and with an objective of determining whether a subset $\Phi \subseteq \mathbf{M}^n$ of streams, based on which we obtain the fused probability $P_\Phi \geq P_{spec}$ of the occurrence/non-occurrence of event, exists under the constraints $C_\Phi \leq C_{spec}$ and $F_\Phi \geq F_{spec}$, where C_Φ and F_Φ are the total cost of using and the overall confidence in subset Φ.

A transformation function $T_r : K \rightarrow T_r(K)$ which maps an instance K of 0-1 KNAPSACK problem into the given instance $T_r(K)$ of **MaxGoal** problem is defined as:

$T_r(\mathbf{U}^n, \mathbf{X}, \mathbf{W}, X_{spec}, W_{spec})$

{

$\mathbf{M}^n = \mathbf{U}^n$,

$\mathbf{P} = \mathbf{X}$,

$\mathbf{F} = NULL$,

$\mathbf{C} = \mathbf{W}$,

$P_{spec} = X_{spec}$,

$C_{spec} = W_{spec}$,

$F_{spec} = 0$,

$\Phi = \Lambda$,

$P_\Phi = X_\Lambda$,

$C_\Phi = W_\Lambda$

}

Note that, relaxing the constraint of confidence (i.e. making $F_{spec} = 0$) reduces the given instance of **MaxGoal** problem into an instance of 0-1 KNAPSACK problem.

We now argue that "K has a solution if and only if $T_r(K)$ has a solution". If a subset Λ of items, with the overall profit X_Λ (by adding the profits obtained from individual items) within the weight $W_\Lambda \leq W_{spec}$, exists in an instance K of the 0-1 KNAPSACK problem; in the corresponding instance $T_r(K)$ of the **MaxGoal** problem, there exists a subset Φ of media streams based on which an overall probability $P_\Phi \geq P_{spec}$ of the occurrence/non-occurrence of event is estimated (by fusing using a Bayesian approach the probabilities of the occurrence/non-occurrence of event based individual streams) within the total cost $C_\Phi \leq C_{spec}$ and with the overall confidence $F_\Phi \geq F_{spec}$. Note that though X_Λ in the 0-1 KNAPSACK problem and P_Φ in **MaxGoal** problem are computed using different methods, but they are equivalent as both are computable in polynomial time and both increase monotonically (as stated in the assumption $\mathcal{A}3$ in section 3.1). We

prove it using Lemma 4.2.3.

It is obvious that the transformation T_r of instances of the two problems can be done in the polynomial time because there is a one-to-one correspondence, and K would have a solution iff $T_r(K)$ has a solution. This proves that the 0-1 KNAPSACK problem is reducible to the **MaxGoal** problem in polynomial time.

Lemma 4.2.3 *The functions to compute the overall profit X_Λ in 0-1 KNAPSACK problem and the overall probability P_Φ in* **MaxGoal** *problem are equivalent.*

Proof: As known, in 0-1 KNAPSACK problem, the function to compute overall profit is additive; whereas, in **MaxGoal** problem, the overall probability of the occurrence/non-occurrence of event is computed using a Bayesian formulation (equation 3.8 in Chapter 3, ignoring the integration of confidence information and the atomic event index j), which is given as -

$P_i = \frac{P_{i-1}.p_i.e^{\overline{\gamma}_i}}{P_{i-1}.p_i.e^{\overline{\gamma}_i}+(1-P_{i-1})(1-p_i).e^{-\overline{\gamma}_i}}$

By making the term $\overline{\gamma}_i = 0$, the above equation becomes -

$= \frac{\rho.\sigma}{\rho.\sigma+(1-\rho)(1-\sigma)}$

where $\rho = P_{i-1}$ and $\sigma = p_i$, and $0 < P_{i-1}, p_i < 1$. This equation, which contains the multiplication and division steps, can easily be transformed to an additive function by replacing the multiplication and division steps with successive additions and subtractions, respectively, as -

$$= \frac{\rho.\sigma}{2.\rho.\sigma+1-\rho-\sigma}$$
$$= \frac{\overbrace{\rho.\rho\ldots}^{\sigma-times}}{2.\overbrace{\rho.\rho\ldots}^{\sigma-times}+1-\rho-\sigma}$$
$$= \frac{\sum_1^\sigma \rho}{2.\sum_1^\sigma \rho+1-\rho-\sigma}$$
$$= \rho'/\sigma'$$

where $\rho' = \sum_1^\sigma \rho$ and $\sigma' = 2.\sum_1^\sigma \rho + 1 - \rho - y$. Note that ρ' and σ' can

be computed in the time of polynomial order $O(d)$, where d is the degree of precision in considering the probability value σ. The further transformation can be done as follows -

$= \sum_1^{\rho'} 1 + (-\sum_1^{\sigma'} 1)$

which are simply additive steps.

The above transformation will also hold for the case when $\overline{\gamma}_i \neq 0$. The only difference would be that the time complexity of computing overall probability using the above equation will be of polynomial order $O(n \times d)$, since the computation of $\overline{\gamma}_i$ (refer to equation 3.11 in Chapter 3) would also require $O(n)$ time.

The above arguments prove Lemma 4.2.3.

Lemma 4.2.4 *Problem* **MaxGoal** *is in NP.*

Proof: To prove that the problem **MaxGoal** is NP, we show that the solution to the decision version of **MaxGoal** problem can be verified in polynomial time.

To verify if there exists a subset Φ of media streams based on which we obtain a fused probability $P_\Phi \geq P_{spec}$ of the occurrence/non-occurrence of event within the total cost $C_\Phi \leq C_{spec}$ and with the overall confidence $F_\Phi \geq F_{spec}$; one can simply make the choices of streams in $O(n)$ time, and can fuse the probabilities (of the occurrence/non-occurrence of event based on individual streams) and their confidence levels. Their costs can simply be added. We can then compare the overall confidence and the total cost of using streams with the specified constraints. If $C_\Phi \leq C_{spec}$ and $F_\Phi \geq F_{spec}$ are true, then the solution is correct, else it is not. This proves that Problem **MaxGoal** does belong to the NP class.

Lemmas 4.2.2, 4.2.3 and 4.2.4 together prove that the Problem **MaxGoal** is NP-Complete.

In the case of problem **MaxConf**, the proof follows the same lines of the reasoning for Problem **MaxGoal**, except that in this case, we would present the same arguments as they are relevant to the F_Φ instead of the P_Φ. Similarly, in the case of problem **MinCost**, the proof follows the same lines, except that in this case, we would present the same arguments for C_Φ instead of the P_Φ. The details are omitted due to space constraints. □

In the light of the Theorem 4.2.1, we develop techniques for obtaining approximate solutions to the problems.

4.3 Developing Approximate Solutions to the MS Problems

From a computational and practical perspective, Theorem 4.2.1 justifies the research for developing heuristic-based solutions, because the optimal solution can only be obtained by an exhaustive search of the entire solution space. The computation of the exact solution by a "brute force" strategy would require a combinatorially explosive number of operations, which is infeasible for typical values of n occurring in any large-scale application. Finally, as mentioned above, there does not seem to be any systematic way by which any partial solution can be discarded except by some type of branch-and-bound philosophy in which a particular subset is discarded (after it is initially investigated) when its current *partial* solution is already more expensive that the *total* solution of another subset.

We develop solutions to these three MS problems **MaxGoal**, **MaxConf** and **MinCost** using the following three heuristics:

$\mathcal{H}1$ In the case of **MaxGoal**, the heuristic is the fused probability of n streams which we quantify as the result obtained from the fusion of

$n-1$ streams and the n^{th} stream, (and the corresponding method of computation utilizing dynamic programming) as explained presently.

$\mathcal{H}2$ In the case of **MaxConf**, the heuristic is the fused *confidence* of n streams, again quantified as the result obtained from the fusion of the confidences of $n-1$ streams and the confidence of the n^{th} stream. Again, the corresponding dynamic programming determines how the latter is computed.

$\mathcal{H}3$ In the case of **MinCost**, the heuristic for n streams is determined as follows. If we select the n^{th}, the best cost would be c_n plus the cost of the approximated optimal solution of using the remaining $n-1$ streams so that the overall probability of the occurrence/non-occurrence of event is at least P_{spec}. However, if we don't select it, then the best cost would possibly be the cost of using the remaining $n-1$ streams.

4.4 Dynamic Programming Based Method

Given the set of n media streams and the system goal (i.e. to detect a compound event $\mathbf{E}_k$) in hand, the solution which approximates the optimal subset of media streams to achieve the system goal is obtained as follows. The compound event $\mathbf{E}_k$ is first decomposed into the atomic events $\mathbf{e}_1$, $\mathbf{e}_2$, $\ldots$, $\mathbf{e}_r$. At a particular time instant, each atomic event $\mathbf{e}_j$, $1 \leq j \leq r$, is detected using *Event Detectors* (ED_{ji}, $1 \leq j \leq r$, $1 \leq i \leq n$), and the steps 1-4 as described in section 3.4.2 are followed. Since the decisions about an atomic event based on different streams could be concurring or contradictory, all the streams are divided into subsets S_1 and S_2; and the overall probabilities $P(\mathbf{e}_j|S_1)$ (of the occurrence) and $P(\bar{\mathbf{e}_j}|S_2)$ (of the non-occurrence) of atomic event $\mathbf{e}_j$ are computed. The overall confidences

F_{S_1} and F_{S_2} in the subsets S_1 and S_2, respectively, are also computed. If $P(\mathbf{e}_j|S_1).F_{S_1} \geq P(\bar{\mathbf{e}_j}|S_2).F_{S_2}$, it is concluded that the atomic event $\mathbf{e}_j$ has occurred and the system finds using dynamic programming based method the optimal subsets Φ while ignoring the subset S_2; otherwise, it is concluded that the event did not occur and the optimal subset is found from S_2, while S_1 is ignored. The system continues to use this optimal subset as long as the system goal is achieved within the specified constraints, otherwise it repeats the whole process of recomputation of the optimal subset. For example, in case of the problem **MaxGoal**, as long as the probability of the occurrence/non-occurrence of event based on the selected optimal subset remains more than a user-specified threshold (i.e. P_{spec}), the same subset is used; else it is recomputed.

In the following three subsections, we describe the dynamic programming based solutions for finding the optimal subset for three different problems **MaxGoal**, **MaxConf** and **MinCost**.

4.4.1 Solution for MaxGoal

In **MaxGoal** problem, the objective is to find a subset $\Phi \in \mathcal{P}(\mathbf{M}^n)$ that maximizes the probability P_Φ of the occurrence/non-occurrence of event subject to $C_\Phi \leq C_{spec}$ and $F_\Phi \geq F_{spec}$. The framework first finds all the subsets Φ_i, $1 \leq i \leq n'$ of streams whose cost $C_{\Phi_i} \leq C_{spec}$, for $1 \leq i \leq n'$; and then, it picks a subset Φ from the subsets Φ_i, $1 \leq i \leq n'$ for which the confidence F_Φ is maximum.

The dynamic programming approach for approximating the optimal subset Φ works as follows. We begin by considering the selection of the n^{th} stream. If we select the n^{th} stream, then the fused probability would be the result obtained from the fusion of n^{th} stream with the remaining $n-1$

streams (with a specified cost $C_{spec} - c_n$, where $c_n < C_{spec}$). However, if we do not select it, the fused probability would possibly be the result obtained from the fusion of the remaining $n-1$ streams (with a specified cost C_{spec}). The fused probability (of the occurrence/non-occurrence of event) will be the maximum of these two possible 'best' options, which also is an integral part of the heuristic function that the solution for **MaxGoal** utilizes.

We thus describe the structure of our solution which converges to the optimal solution by the following recurrence relation:

$$Prob(i,m) = \begin{cases} Prob(i-1,m), & c_i > m \\ max[Prob(i-1,m), \mathbf{PFusion}(Prob(i-1,m-c_i), & \\ \qquad p_i, \Gamma)] & c_i \leq m \end{cases}$$

where $Prob(i,m)$, $1 \leq i \leq n$, $1 \leq m \leq C_{spec}$, approximates the optimal fused probability (of the occurrence/non-occurrence of event) based on streams 1 to i with the cost m. The initial conditions for the recursive relation are:

$$Prob(1,m) = \begin{cases} 0 & c_1 > m \\ p_1 & c_1 \leq m \end{cases}$$

The **PFusion** function combines the probabilities of the occurrence/non-occurrence of event based on two sources $\mathbf{M}^{i-1}$ and M_i using the assimilation model given in equation (3.8) (Refer to section 3.4.2 in Chapter 3).

The optimal fused probability is approximated by recursively computing $Prob(n,m)$. As soon as the $Prob$ table is constructed, the proposed solution, which approximates the optimal subset Φ is computed by backtracking through the table.

The algorithm **MaxGoal** outlines the idea described above.

MaxGoal(n, p, Γ, c, f, C_{spec}, F_{spec})

Inputs

n : Number of input media streams.

$p[1 \ldots n]$: Probability of the occurrence/non-occurrence of event based on each stream.

$f[1 \ldots n]$: Confidence in each media stream.

$c[1 \ldots n]$: Cost of using each media stream.

Γ: Set of agreement coefficients among media streams.

C_{spec} : Specified maximum cost.

F_{spec} : Specified minimum confidence.

Steps

1. Initialize $Prob$, $Conf$ and $Select$ array to zero.

2. for $i = 1$ to n, $m = 0$ to C_{spec}

3. if $(c[i] \leq m)$

4. Compute fused probability P_i using equation (3.8)

5. Compute overall confidence F_i using equation (3.13)

6. if $(P_i > Prob[i-1, m])$ $Prob[i, m] = P_i$, $Conf[i, m] = F_i$, $Select[i, m] = 1$

7. else $Prob[i, m] = Prob[i-1, m]$, $Conf[i, m] = Conf[i-1, m]$, $Select[i, m] = 0$

8. else $Prob[i, m] = Prob[i-1, m]$, $Conf[i, m] = Conf[i-1, m]$, $Select[i, m] = 0$

9. $kk = m - 1$, $P_\Phi = Prob[n, kk]$, $C_\Phi = 0$

10. for $i = n$ to 1 in steps -1

11. if $(Select[kk] == 1)$

12. Output the stream i into Φ

13. $C_\Phi = C_\Phi + c[i]$, $kk = kk - c[i]$

14. F_Φ = maximum confidence at C_Φ

Outputs

P_Φ: An approximation to the optimal probability of the occurrence of event based on subset Φ.

Φ: The set of media streams used to obtain P_Φ.

C_Φ: The overall cost of using Φ to obtain P_Φ.

F_Φ: The overall confidence in subset Φ.

4.4.2 Solution for MaxConf

The problem **MaxConf** is similar to problem **MaxGoal** except that in **MaxConf** problem, the objective is to maximize the overall confidence in the selected subset; while in **MaxGoal**, we maximize the overall probability of the occurrence/non-occurrence of event.

Similar to problem **MaxGoal**, for problem **MaxConf**, we first find all the subsets Φ_i, $1 \leq i \leq n'$ of streams whose cost $C_{\Phi_i} \leq C_{spec}$, for $1 \leq i \leq n'$. Then, we pick a subset Φ from the subsets Φ_i, $1 \leq i \leq n'$ for which the overall probability P_Φ of the occurrence/non-occurrence of event is maximum.

The dynamic programming solution for **MaxConf** works as follows. We approximate the optimal solution by the following recurrence relation given as:

$$Conf(i,m) = \begin{cases} Conf(i-1,m), & c_i > m \\ max[Conf(i-1,m), \textbf{CFusion}(Conf(i-1, & \\ \qquad m-c_i), f_i)] & c_i \leq m \end{cases}$$

where $Conf(i,m)$, $1 \leq i \leq n$, $1 \leq m \leq C_{spec}$, approximates the optimal overall confidence in the streams 1 to i with the cost m, and is the "local" heuristic function that **MaxConf** resorts to. The initial conditions for the recursive relation are -

$$Conf(1,m) = \begin{cases} 0 & c_1 > m \\ f_1 & c_1 \leq m \end{cases}$$

The **CFusion** combines the confidence levels in two sources $\mathbf{M}^{i-1}$ and M_i using the fusion model given in equation (3.13). We approximate the optimal overall confidence by recursively computing $Conf(n,m)$. Once the

$Conf$ table is constructed, the reported solution, which is the approximation to the optimal subset, Φ, is found by backtracking through the table.

Similar to **MaxGoal**, the algorithm **MaxConf** can be outlined as given below.

MaxConf(n, p, Γ, c, f, C_{spec}, F_{spec})

Inputs

n, p, c, f, Γ, C_{spec} and F_{spec}: Similar to **MaxGoal**

Steps

1. Initialize $Conf$, $Prob$ and $Select$ array to zero.
2. for $i = 1$ to n, $m = 0$ to C_{spec}
3. if ($c[i] \leq m$)
4. Compute overall confidence F_i using equation (3.13)
5. Compute fused probability P_i using equation (3.8)
6. if ($F_i > Conf[i-1, m]$) $Conf[i, m] = F_i$, $Prob[i, m] = P_i$, $Select[i, m] = 1$
7. else $Conf[i, m] = Conf[i-1, m]$, $Prob[i, m] = Prob[i-1, m]$, $Select[i, m] = 0$
8. else $Conf[i, m] = Conf[i-1, m]$, $Prob[i, m] = Prob[i-1, m]$, $Select[i, m] = 0$
9. $kk = m - 1$, $F_\Phi = Conf[n, kk]$, $C_\Phi = 0$
10. for $i = n$ to 1 in steps -1
11. if ($Select[kk] == 1$)
12. Output the stream i into Φ
13. $C_\Phi = C_\Phi + c[i]$, $kk = kk - c[i]$
14. F_Φ = maximum confidence at C_Φ

Outputs

F_Φ: An approximation to the optimal confidence obtained.

Φ: The set of media streams used to obtain F_Φ.

C_Φ: The overall cost of using Φ to obtain F_Φ.

P_Φ: The overall probability of the occurrence of event based on subset Φ.

4.4.3 Solution for MinCost

The problem **MinCost** is different from **MaxGoal** and **MaxConf** in that the optimization functions in **MaxGoal** and **MaxConf** are to *maximize* probability and confidence, respectively; while in **MinCost**, we *minimize* the cost.

We first find all the subsets Φ_i, $1 \leq i \leq n'$ of streams whose fused probabilities $P_{\Phi_i} \geq P_{spec}$, for $1 \leq i \leq n'$. Then, we pick a subset Φ from the subsets Φ_i, $1 \leq i \leq n'$ for which the confidence F_Φ is maximum.

To solve **MinCost** using a dynamic programming approach, we begin by considering the n^{th} stream. If we select it, the best cost would be c_n plus the cost of the approximated optimal solution of using the remaining $n-1$ streams so that the overall probability of the occurrence/non-occurrence of event is at least P_{spec}. However, if we don't select it, then the best cost would possibly be the cost of using the remaining $n-1$ streams. The approximate to the optimal cost of determining the occurrence/non-occurrence of event will be the minimum of these two "best" options, and this will be the heuristic function that **MinCost** depends on so as to invoke the results of [64].

Let $Cost(i, m)$ denote the cost of using media stream $1 \ldots i$ for achieving the goal with probability m. Assuming that probability takes one of the L discrete values, we characterize the recursive relation for $Cost(i, m)$ as

follows -

$$Cost(i,m) = \begin{cases} min(Cost(i-1,m), c_i) & , m \leq min(p_i, P_{spec}) \\ \textbf{while}(l[ss] \neq 0) & \\ \{ & \\ \quad min(Cost(i,m), fcost) & p_i < m \leq R \textbf{ and } Cost(i,m) \neq \infty \\ \quad min(Cost(i-1,m), fcost) & p_i < m \leq R \textbf{ and } Cost(i,m) = \infty \\ \} & \\ Cost(i-1,m) & m > R' \end{cases}$$

where $1 \leq i \leq n$, $1 \leq m \leq L$. The initial conditions are:

$$Cost(1,m) = \begin{cases} c_1 & , m \leq min(p_1, P_{spec}) \\ \infty & , m > p_1 \end{cases}$$

In the recursive formulation described above, $fcost$, R and R' are computed as,

$$fcost = \begin{cases} Cost(i-1, l[ss]) & , ss > 0 \textbf{ and } l[ss] \neq p_i \\ c_i & , ss > 0 \textbf{ and } l[ss] = p_i \\ 0 & , ss = 0 \end{cases}$$

$$R = \begin{cases} \textbf{PFusion}(l[ss], p_i) & , ss > 0 \textbf{ and } l[ss] \neq p_i \\ p_i & , ss > 0 \textbf{ and } l[ss] = p_i \\ 0 & , ss = 0 \end{cases}$$

$$R' = \begin{cases} max(R', R) & , ss > 0 \\ 0 & , ss = 0 \end{cases}$$

The $l[ss]$ is an array that contains the probabilities based on the individual streams, as well as the fusion probabilities. After constructing the $Cost$ table, the $Select$ array is traced back to find the solution which approximates the optimal subset, Φ.

The algorithm **MinCost** is given as follows.

MinCost(n, p, c, f, Γ, L, P_{spec}, F_{spec},)

Input

n, p, c, f, Γ and F_{spec}: Similar to **MaxGoal**

L: Number of discrete levels of probability values

$P_{spec} \leq L$: Specified minimum fused probability of achieving the goal

Steps

1. Initialize $Cost$ to ∞, L to 100, and $Prob$, $Conf$ and $Select$ array to zero.

2. for $i = 0$ to n

3. for $m = 0$ to $Min(p_i, L)$

4. $Cost[i, m] = Min(Cost[i-1, m], c_i)$

5. if $(Cost[i, m] == Cost[i-1, m])$

6. $Conf[i, m] = Conf[i-1, m]$, $Prob[i, m] = Prob[i-1, m]$, $Select[i, m] = 0$

7. else $Conf[i, m] = f_i$, $Prob[i, m] = p_i$, $Select[i, m] = 1$

8. Initialize variables $R = R' = 0$, $ss = 0$, $fcost = 0$, $fconf = 0$, $fprob = 0$

9. ss = Number of unique values in $Cost$ array, copy them into l array

10. while $(l[ss] \neq 0)$

11. if $(l[ss] \neq p_i)$

12. $fprob$ =**PFusion**$(l[ss], p_i, \Gamma)$

13. $fconf$ =**CFusion**$(l[ss], f_i)$

14. $fcost = Cost[i-1, l[ss]] + c_i$

15. else $fprob = p_i$, $fconf = f_i$, $fcost = c_i$

16. $R = fprob$

17. for $m = m'$ to R

18. if $(Cost[i, m] \neq \infty)$ $Cost[i, m] = min(Cost[i, m], fcost)$

19. if $(Cost[i, m] == fcost)$ $Conf[i, m] = f_i$, $Prob[i, m] = p_i$

20. else $Cost[i, m] = min(Cost[i-1, m], fcost)$

21. if $(Cost[i,m] == fcost)$ $Conf[i,m] = f_i$, $Prob[i,m] = p_i$

22. else $Conf[i,m] = Conf[i-1,m]$, $Prob[i,m] = Prob[i-1,m]$

23. if $(Cost[i,m] \neq Cost[i-1,m]$ and $Cost[i,m] \neq \infty)$ $Select[i,m] = 1$

24. else $Select[i,m] = 0$

25. $m' = R+1$, $R' = max(R', R)$, $ss = ss+1$

26. for $m = R'+1$ to L

27. $Cost[i,m] = Cost[i-1,m]$, $Conf[i,m] = Conf[i-1,m]$, $Prob[i,m] = Prob[i-1,m]$

28. $Select[i,m] = 0$

29. $OptProb = P_{spec}$

30. if $(OptProb < L)$

31. while $(Cost[i, OptProb+1] == Cost[i, OptProb])$ $OptProb = OptProb - 1$

32. else

33. while $(Cost[i, OptProb] == Cost[i, OptProb-1])$ $OptProb = OptProb - 1$

34. $OptProb = OptProb - 1$

35. $P_\Phi = OptProb$, $C_\Phi = 0$, $i = i-1$, $m = OptProb$, $C_\Phi = kk = Cost[i, OptProb-1]$

36. while $(kk > 0)$

37. while $(Cost[i,m] \neq kk)$ $m = m-1$

38. if $(Select[i,m] == 1)$ Output i into Φ, $kk = kk - c_i$

39. $i = i-1$

40. F_Φ = maximum confidence at P_Φ

Outputs

Φ: The set of media streams used whose cost is C_Φ.

C_Φ: An approximation to the optimal cost of using Φ to obtain P_Φ.

P_Φ: The overall probability of the occurrence of event based on subset Φ.

F_Φ: The overall confidence in the subset Φ.

4.5 Complexity Analysis

Any brute-force approach to solve each of the three problems **MaxGoal**, **MaxConf** and **MinCost** requires $O(2^n)$ time since all the 2^n combinations of streams need be checked to find the optimal subset. We have also proven these three MS problems to be NP-Complete in section 4.2. However, the proposed dynamic programming based approach solves them in *pseudo*-polynomial time. We call it *pseudo*-polynomial time-complexity because it is the polynomial time-complexity under the following assumptions -

- The total cost of media streams is not exponential in terms of total number of media streams, i.e. $C_n \neq O(2^n)$ (for problems **MaxGoal** and **MaxConf**).
- The total discrete levels L of probability values is not exponential in terms of total number of media streams, i.e. $L \neq O(2^n)$ (for problem **MinCost**).

The time complexity of both **MaxGoal** and **MaxConf** algorithms is $O(n^2 \times C_{spec})$, where $C_{spec} \leq C_n$. This is on average lower than of the brute-force approach. Note that $O(n^2 \times C_{spec})$ also includes the time complexity of **PFusion**, which is $O(n)$. The space complexity of the **MaxGoal** algorithm is $O(n \times C_{spec})$.

The algorithm **MinCost** has a time complexity of $O(n^2 \times L)$ to approximate the optimal subset which is again better than the brute-force approach. Note that higher the discrete levels L of probability value, higher the time complexity would be. In the algorithm **MinCost**, we have used $L = 100$. The space complexity is $O(n \times P_{spec})$, where $P_{spec} \leq L$.

4.6 Simulation Results

In this section, we provide the simulation results to show the utility of the proposed dynamic programming based method for obtaining the optimal subset of streams. In simulation results, we show the tradeoff only between the probability with which the goal is achieved (in other words, the probability of the occurrence/non-occurrence of event) and the overall cost of using the streams [2]. We do not consider confidence information here. Note that, the experimental results on real data will be provided in Chapter 5, section 5.3, where we will show a three-fold tradeoff among - the probability with which the goal is achieved, the overall cost of using streams, and the overall confidence in streams.

We consider a system with 10 media streams. The individual probabilities of the occurrence of an event (say $\mathbf{E}_k$) based on them and their cost are given by arrays $p = (0.70, 0.45, 0.65, 0.40, 0.75, 0.45, 0.85, 0.30, 0.55, 0.60)$ and $c = (9, 9, 4, 2, 8, 2, 8, 5, 2, 3)$, respectively. First, the streams are divided into two sets S_1 and S_2 based on whether the system obtains concurring or contradictory evidences using them. Precisely, the streams based on which the system obtains the probability of the occurrence of event more than 0.50 are put in set S_1 and rest in set S_2. So, we get $S_1 =$ (0.70, 0.65, 0.75, 0.85, 0.55, 0.60) and $S_2 =$ (0.55, 0.60, 0.55, 0.70). The probability values in S_2 have been computed by complementing the probabilities of the occurrence of the event. Note that, after this division, the sets S_1 and S_2 of streams support the occurrence and non-occurrence of the event, respectively. Next, we assimilate the streams from two sets individually and obtain the fusion probabilities $P(\mathbf{E}_k|S_1)$ and $P(\bar{\mathbf{E}_k}|S_2)$ (Refer to Table 4.1). In simulation results, we have assumed uniform agreement coefficient among all the media streams for sake of simplicity. However, we analyze how the system behaves

Table 4.1: Fusion probabilities of S_1 and S_2

Agreement coefficient	0	0.50	1.00
$P(\mathbf{E}_k\|S_1)$	0.9927	1.0000	1.0000
$P(\bar{\mathbf{E}}_k\|S_2)$	0.8394	0.9906	0.9995

by having different values (0.00, 0.50 and 1.00) of this uniform agreement coefficient. As shown in Table 4.1, $P(\mathbf{E}_k|S_1)$ is higher than $P(\bar{\mathbf{E}}_k|S_2)$; this implies the occurrence of event. So, we find the optimal subset from S_1 using **MaxGoal** and ignore the set S_2.

We study the behavior of **MaxGoal** and **MinCost** by varying the specified maximum cost C_{spec} and the specified minimum probability P_{spec} of achieving the goal, respectively. The simulation results of **MaxGoal** and **MinCost** are shown in figure 4.1a-4.1b and figure 4.1c-4.1d, respectively. In figure 4.1a-4.1d, symbols **A**, **B**, and so on, represent the optimal subsets. For instance, in figure 4.1b, symbol **B** (i.e. $\Phi = (2, 3)$) represents a subset of 2^{nd} and 3^{rd} stream of S_2 set. The x-axis value corresponding to $\Phi = (2, 3)$ shows the cost $C_\Phi = 4$ of using the subset Φ and y-axis shows the optimal probability $P_\Phi = 0.9313$ achieved by using this subset. Note that the symbol **B** indicates the optimal subset obtained by having the uniform agreement coefficient as 1.00. Also note that the same subset Φ with the same cost C_Φ achieves a lower probability when the agreement coefficient between the streams is low (the symbols **C** and **D**).

The overall observations from simulation (figure 4.1) are -

1. The proposed dynamic programming based method offers a flexibility to compare whether any one set of media streams of low cost would be better than any other set of media streams of higher cost. For instance, figure 4.1a clearly shows that the subset indicated by symbol **E** would be a better choice over the subset indicated by symbols **H** onwards

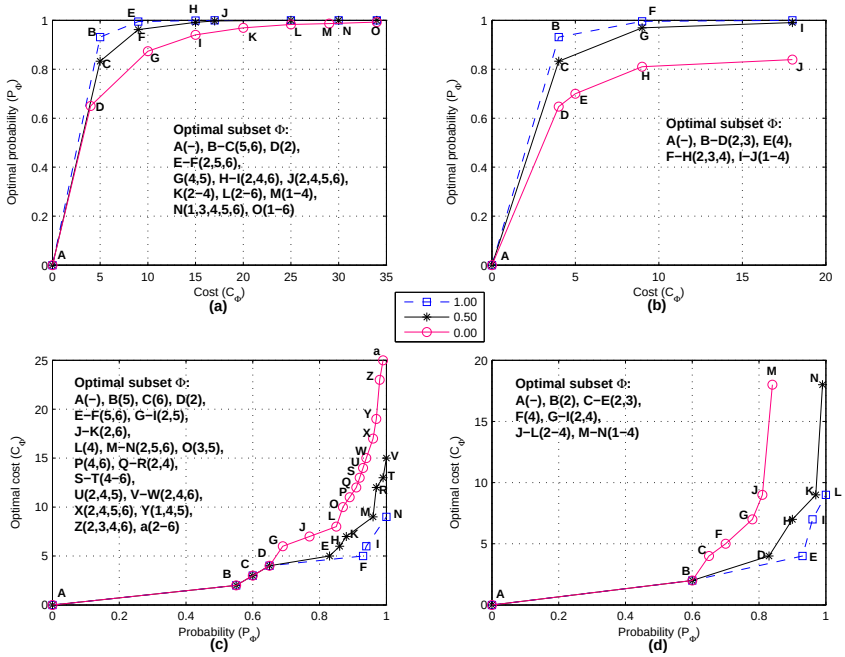

Figure 4.1: Simulation results: (a) **MaxGoal** on S_1, (b) **MaxGoal** on S_2, (c) **MinCost** on S_1 and (d) **MinCost** on S_2. The legends show the varying value of agreement coefficient.

since there is a very small difference in the goal achieved using the two subsets while there is a significant difference in the cost.

2. The graphs (figure 4.1) show a pictorial representation of which subset of streams is most suitable in terms of optimal probability or the optimal cost. It also helps in deciding which is next best subset of streams in case the best subset is not available. For instance, in figure 4.1c, if the subset denoted by **O** is not available then next best subset (in terms of cost) denoted by **P** can be considered for use.

3. Fewer streams with high agreement among them are more advantageous (in terms of cost and fusion probability) compared to using more streams with lower agreement. For example, in figure 4.1a, the subset denoted by **H** having a higher agreement coefficient (i.e. 1.0) among

its streams provides a higher fused probability value compared to a subset denoted by **I** which has a lower agreement cofficient (i.e. zero) among its streams. Similarly, in figure 4.1c, the subset denoted by **N** which has a higher agreement coefficient (i.e. 1.0) among its streams is able to make a decision at a lower cost (8 vs. 15) compared to a subset denoted by **V** whose streams have a lower agreement coefficient (i.e. 0.50) among them.

Chapter 5

Experiments and Evaluation

In this chapter, we present the experimental results to demonstrate the utility of the proposed framework for information assimilation, and also evaluate its performance with and without using - agreement/disagreement information and the confidence information in streams. This chapter begins with a brief description of surveillance system, which we have implemented, in section 5.1. Then, the results are presented in two parts - first, we present the information assimilation results in section 5.2; and next, the results for optimal subset selection are provided in section 5.3.

5.1 System Description

The surveillance environment is the corridor of our school building and the system goal is to detect events that are described in Example 3.1 (in section 3.2 of Chapter 3) i.e. human's running, walking, standing, talking, shouting and door knocking in the corridor. The environment layout is shown in figure 5.1. We use two video sensors (Canon VC-C50i cameras denoted by M_1 and M_2) to record the video from the two opposite ends of corridor, and two audio sensors (USB microphones denoted by M_3 and M_4) to capture

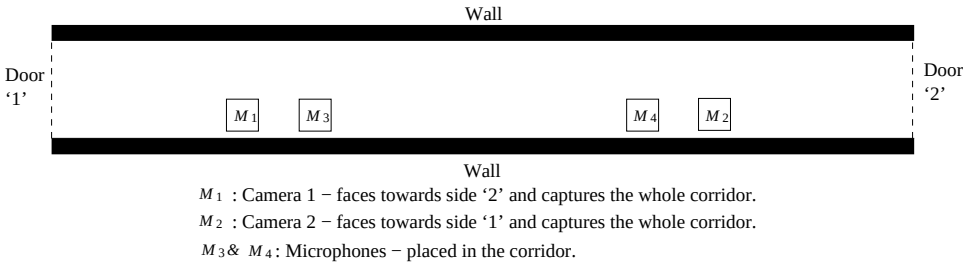

Figure 5.1: The layout of the corridor under surveillance and monitoring

the ambient sound. The two cameras and two microphones are connected to a central PC (Pentium-IV 3.6 GHz), as shown in figure 5.2. A Picolo-Pro video capture card is used to capture the image data.

A software interface application has been developed for performing various system operations such as for recording and processing of data, for submitting queries, and for evaluating system performance. A snapshot of the multimedia surveillance system which we have developed is shown in figure 5.3. The system is implemented using Visual C++ on the MS-Windows platform. MS-Access is used as the database to store the features and the events. Note that our system works on the recorded data. Realtime implementation of the proposed framework would encounter several difficulties such as realtime processing of streams, synchronization of heterogeneous streams etc. This thesis does not claim to address them.

5.2 Information Assimilation Results

In this section, we present the results for information assimilation and show how the framework performs better by using sensors' two properties - agreement coefficient and confidence information [4]. We provide the data set used for the experiments in subsection 5.2.1. The performance evaluation

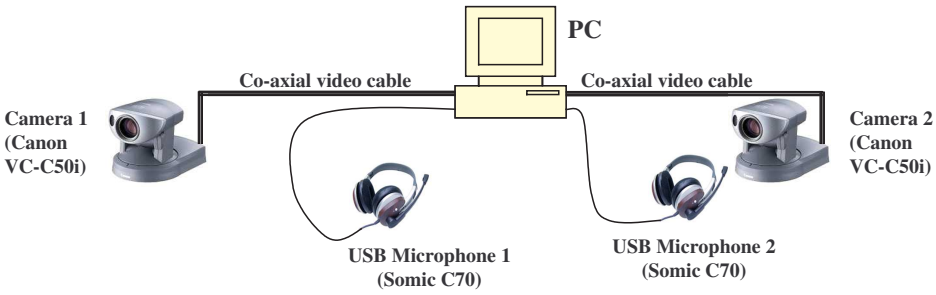

Figure 5.2: System setup

Figure 5.3: Multimedia Surveillance System

criteria are stated in subsection 5.2.2. In subsection 5.2.3, we describe the preprocessing steps performed on the video and audio data in order to detect events. We present an illustrative example, in subsection 5.2.4, to show how our proposed framework works in order to detect an event over a timeline. Finally, in subsection 5.2.5, we present the overall performance analysis to demonstrate the utility of the proposed framework.

5.2.1 Data set

For our experiments, we have used data of more than twelve hours which has been recorded using the system consisting of two video cameras and two USB microphones (as described in section 5.1) in the corridor of our school building. Over the period of more than twelve hours, a total of 92 events occurred over for a period of 1268 seconds. The details of various events and their time durations are given in Table 5.1. The graduate students from our lab volunteered to perform these activities. The images of some of the captured events are shown in figure 5.4.

5.2.2 Performance evaluation criteria

The evaluation of proposed framework is performed based on two tasks - event detection and event classification. The evaluation of event detection task is characterized by two metrics - False Rejection Rate (FRR) and False Acceptance Rate (FAR), which are defined as follows -

$$\text{FRR} = \frac{\text{Number of events not detected}}{\text{Total number of events}}$$

$$\text{FAR} = \frac{\text{Number of non-events detected}}{\text{Total number of non-events}}$$

The event classification task is evaluated based on the accuracy (ACC)

Figure 5.4: The images of some of the captured events: (a) Walking (b) Running (c) Standing and Talking (d) Walking and Talking (e) Door knocking (f) Standing and Shouting

Table 5.1: The data set

Events	Time duration (In seconds)
Standing	139
Walking	798
Running	142
Standing, Talking	30
Standing, Shouting	11
Standing, Knocking	59
Walking, Talking	80
Walking, Shouting	9

in classification. The metric ACC is defined as follows -

$$\text{ACC} = \frac{\text{Number of events correctly classified}}{\text{Total number of events that are detected to be the valid events}}$$

An event here refers to the observation made over a t_w time period (Refer to section 3.3).

As described in section 3.3, it is critical to determine the value of t_w. We have determined through experiments the suitable value of t_w to be 1 second for our data set. Note that t_w here implies the granularity of observations. As can be seen from figure 5.5, at $t_w = 1$ second, we obtain the maximum accuracy (ACC) and minimum FRR.

5.2.3 Preprocessing steps

Event detection in video streams

The video is processed to detect human motion (running, walking and standing). Video processing involves two major steps - background modeling and blob detection. The background is modeled using an adaptive Gaussian method [79, 47]. The blob detection is performed by first segmenting the foreground from the background using simple 'matching' on the three RGB color channels, and then using the morphological operations (erode and dila-

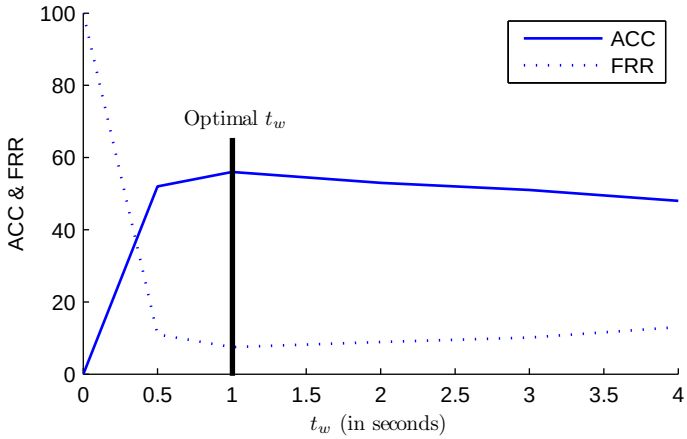

Figure 5.5: Determining the optimal value of t_w

tion) to obtain connected components (i.e. blobs). The matching is defined as a pixel value being within 2.5 standard deviations of the distribution. We have also explored the use of experiential sampling technique for improving the efficiency of the process of foreground/background segmentation [6]. However, since it is not the main focus of this thesis, we do not report the corresponding results.

A summary of the video features used for various classification tasks is provided in Table 5.5(a). We assume that the blob of an area greater than a threshold corresponds to a human. The detected blob and its bounding rectangle is shown in figure 5.6. Once we compute the bounding rectangle (x, y, w, h) for each blob, where (x, y) denotes the top-left coordinate, w is the width and h is the height; we map the point $(x + w/2, h)$ (i.e. approximating with human's feet) in the image to a point (Ex, Ey) in 3-D world (i.e. on the corridor's floor), as shown in figure 5.7. To achieve this mapping, we calibrate the cameras and obtain a transformation matrix that maps image points to the points on corridor's floor. This provides the exact

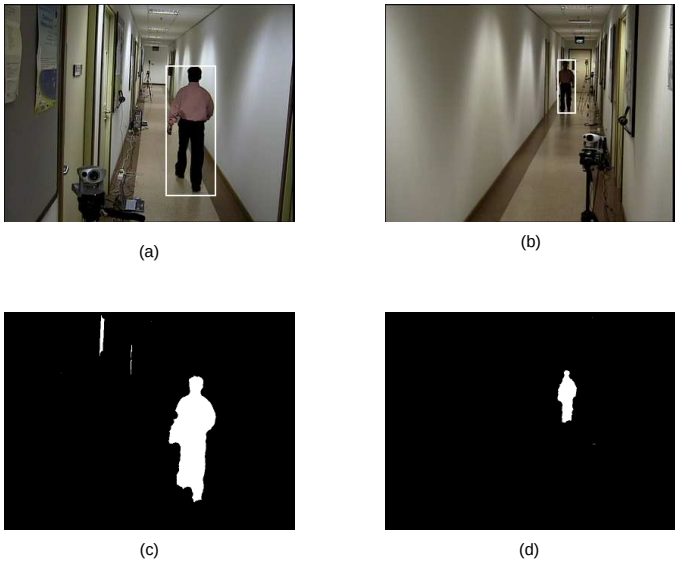

Figure 5.6: Blob detection in Camera 1 and Camera 2: (a)-(b) Bounding rectangle, (c)-(d) Detected blobs

ground location of the human in the corridor at a particular time instant.

The system identifies the start and end of an event in video streams as follows. If a person moves towards the camera, the start of event is marked when the blob's area becomes greater than a threshold and the event is considered as ended when the blob intersects with the boundary of the image. However, if the person walks away from the camera, the start and end of the event is inverted. The event detection is performed at regular time intervals of $t_w = 1$ second. Using the actual location of the person on the corridor's ground at the end of each time interval t_w, we compute the *average distance* traveled by a person on the ground. The average distance instead of the actual traveled distance is considered to minimize the effect of errors in blob detection. Based on this average distance, a Bayes classifier is first trained and then used to classify an atomic event to be one of the

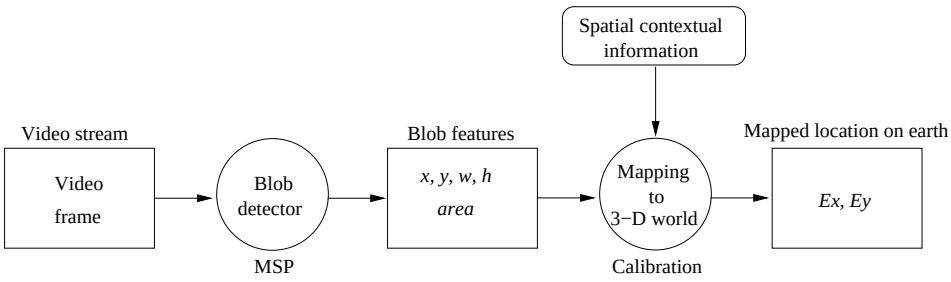

Figure 5.7: The process of finding from a video frame the location of a person on the corridor ground in 3-D world

classes - standing, walking and running. The Bayesian classifier provides the probabilistic decision about an event.

Event detection in audio streams

Using the audio streams, the system detects events such footsteps, talking, shouting and door knocking. The audio (of 44.1 MHz frequency) is divided into the "audio frames" of 50 ms each. The frame size is chosen by experimentally observing that 50 ms is the minimum period during which an event such as a footstep can be represented. We adopted a hierarchical (top-down) approach to model these events using a mixture of Gaussian (GMM). The top-down event modeling approach works better than compared to the single-level multi-class modeling approach. We performed a separate study to find the suitability of features for detecting these audio events [7]. Again, since this is not the main focus of thesis, we have not reported here the corresponding results.

Table 5.5(b) summarizes the audio features used for foreground/ background segmentation and for classification of events at different levels. The feature Log Frequency Cepstral Coefficients (LFCCs) with 10 coefficients and 20 filters worked well for foreground/background segmentation and for

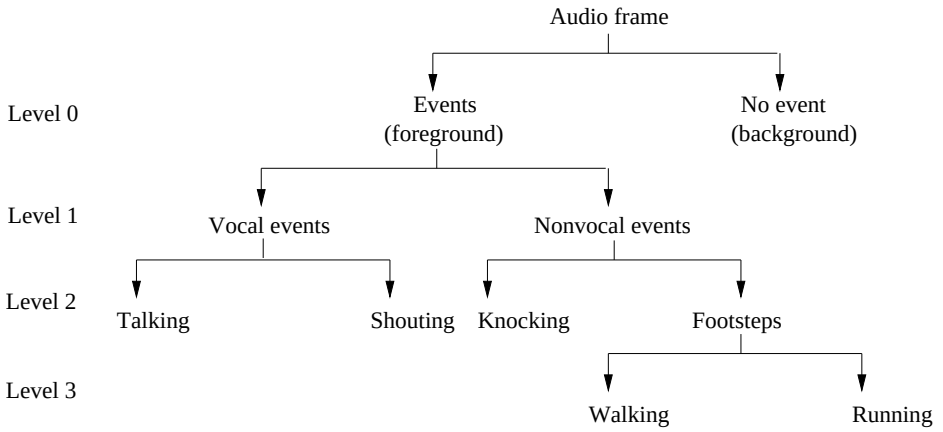

Figure 5.8: Audio event classification

distinguishing between vocal/nonvocal and footsteps/knocking events. The LFCCs are computed by using logarithmic filter bank in frequency domain [55]. The Linear Predictor Coefficient (LPC) that have been widely used in speech processing community worked well for demarcating between talking and shouting events.

The Gaussian Mixture Model (GMM) classifier is employed to classify every audio frame (of 50 ms) into the audio events at different levels as shown in figure 5.8. At the top level (0), each input audio frame is classified as the foreground or the background. The background is the environment noise which represents 'no event' and is ignored. The foreground that represents the events, are further categorized into two classes - vocal and nonvocal (level 1). At the next level (2), both vocal and nonvocal events are further classified into "talking/shouting" and the "footsteps/door knocking" events, respectively. Finally, at the last level (3), the footsteps sequences are classified as "walking" or "running" based on the frequency of their occurrence in a specified time interval.

Similar to the video, the system makes a probabilistic decision about the events based on audio streams after every $t_w = 1$ second. Note that, in 1

Table 5.2: A summary of the features used for various classification tasks in video and audio streams

(a) Video

Classification task	Feature used
Foreground/Background	RGB channels
Running/Walking/Standing	Blob's displacement

(b) Audio

Classification task	Feature used
Foreground/Background	LFCC
Vocal/Nonvocal	LFCC
Talk/Shout	LPC
Footsteps/Door knocking	LFCC

second, we obtain 20 audio frames of 50 ms each. The audio event classification for the audio data of t_w time period is performed as follows. First, the system learns via training the number of audio frames corresponding to an event in the audio data of t_w time period. Then, a Bayesian classifier is employed to estimate the probability of occurrence of an audio event at a regular time interval t_w.

5.2.4 Illustrative example

In this section, we describe with an example how the proposed framework works in order to detect an event over a timeline. Let us consider a compound event $\mathbf{E}_k$ "A person is walking, knocking the door and then continued walking in the corridor". This event consists of atomic events occurring in two different ways. First, it consists of two atomic events occurring together i.e. "standing" and "door knocking" events. Second, it also consists of atomic events occurring one after another i.e. "walking" event followed by "standing/door knocking" event and then followed by "walking" event. The audio data captured using microphone 1 and microphone 2 is shown in shown in figure 5.9. Figure 5.9 distinctly shows the "door knocking" events.

Some of the video frames captured by camera 1 and camera 2 corre-

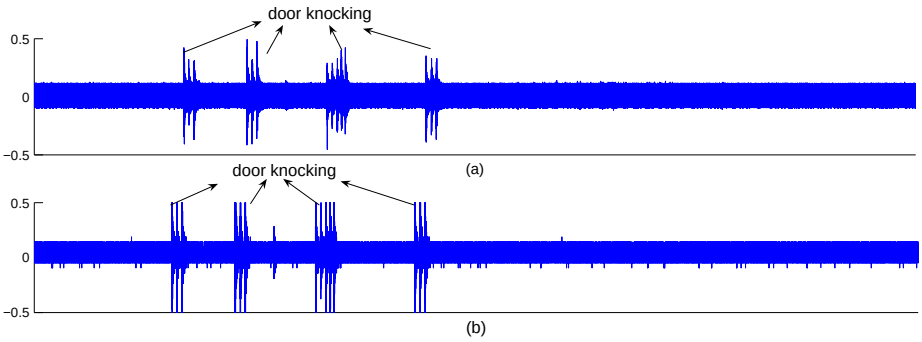

Figure 5.9: Audio data captured by (a) microphone 1 and (b) microphone 2 corresponding to the event $\mathbf{E}_k$

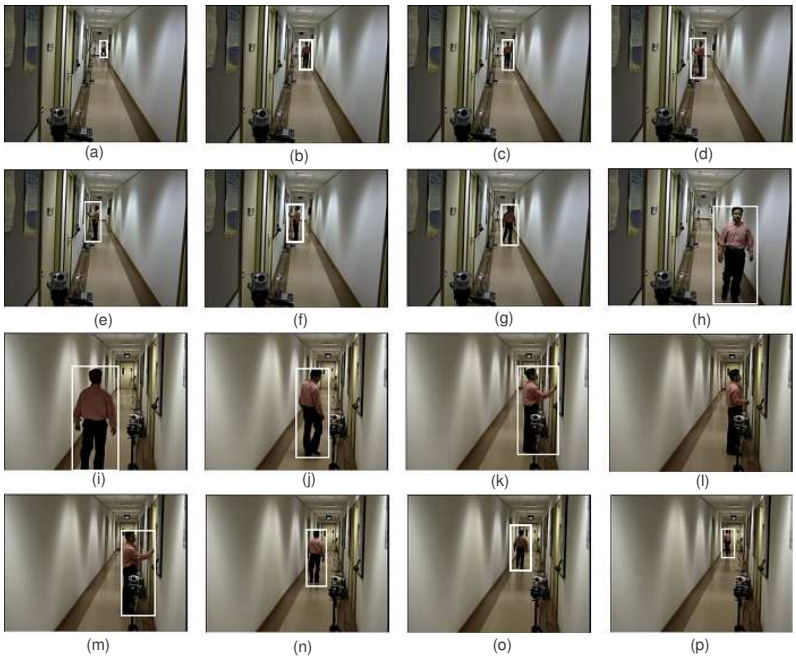

Figure 5.10: Some of the video frames captured by (a)-(h) camera 1 and (i)-(p) camera 2 corresponding to the event $\mathbf{E}_k$.

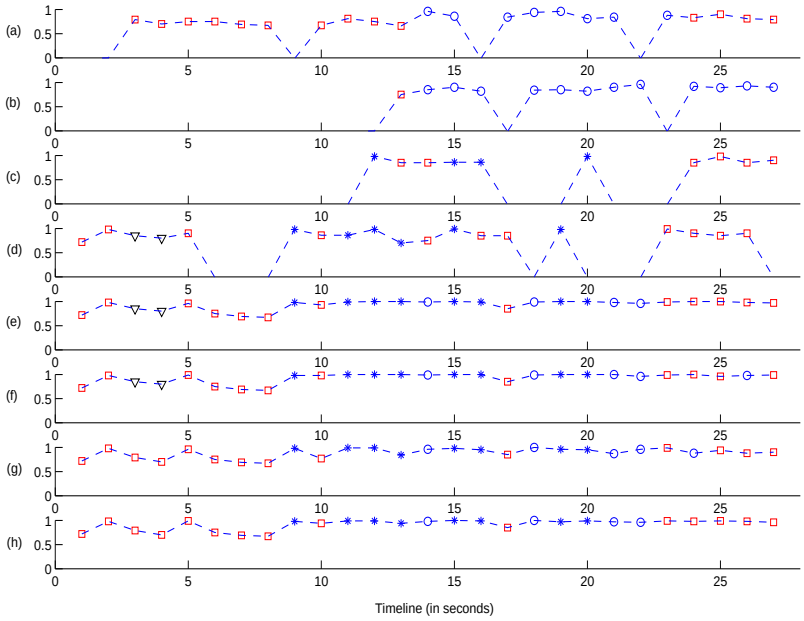

Figure 5.11: Timeline-based assimilation of probabilistic decisions about the event $\mathbf{E}_k$. The legends denote the probabilistic decisions based on (a) Video stream 1 (b) Video stream 2 (c) Audio stream 1 (d) Audio stream 2 (e) All the streams (without agreement coefficient and confidence information) (f) All the streams (with agreement coefficient but without confidence information) (g) All the streams (with confidence information but without agreement coefficient) (h) All the streams (with both agreement coefficient and the confidence information)

sponding to the event $\mathbf{E}_k$ and the bounding rectangles of the detected blobs in them are shown in figure 5.10. The camera 1 images labeled by (a)-(c), (g)-(h) show the "walking" event; and the images labeled by (d)-(e) show the "door knocking" event. Similarly, in camera 2, images labeled by (i)-(j), (n)-(p) show the "walking" event; and the images labeled by (k)-(m) show the "door knocking" event.

The system detects the walking event using both audio and video streams, while standing and knocking events are detected based on video and audio streams, respectively. The probabilistic decisions about these atomic events

are obtained based on respective streams at every $t_w = 1$ second. The overall decision for compound events are obtained along the timeline by assimilating the probabilistic decisions for atomic events as shown in figure 5.11. Note that in figure 5.11, the legends denote as follows: '$\circ$' - "standing", '$\square$' - "walking", '$\bigtriangledown$' - "running" and '$*$' - "door knocking" events.

Figures 5.11a-5.11d show the timeline-based probabilistic decisions based on individual streams. Figures 5.11e-5.11h show the combined decision about the event at a regular time interval with and without using streams' agreement/disagreement and confidence information.

It is interesting to note from figure 5.11 that though using agreement coefficient improves the accuracy of computing the probability of occurrence of an event, it is also important to use the confidence information to avoid incorrect results. For instance, using the stream's confidence information helps in obtaining correct results at time instants 3 and 4 in figure 5.11g-5.11h as compared to the results at the same time instants in figure 5.11e-5.11f where confidence information is not used and an "walking" event is detected as "running". Note that the correct sequence of event is as follows: Time instants 1-9 "walking", 10-20 "standing/door knocking" and 21-27 "walking".

5.2.5 Overall performance analysis

Using Individual Streams

First, we performed event detection and classification using individual streams. The probability threshold Th value for determining the occurrence of an event was set to 0.70. The probability threshold Th is a threshold to convert a probabilistic decision into a binary decision (Refer to section 3.4.3). We have also investigated the effect of varying the probability threshold Th

Table 5.3: Results: Using individual streams with $Th = 0.70$

Stream	FRR	FAR	ACC
Video stream 1	0.12	0.01	0.60
Video stream 2	0.10	0.03	0.60
Audio stream 1	0.07	0.19	0.55
Audio stream 2	0.06	0.27	0.51

(from 0.50 to 0.99) onto the accuracy of event detection. It is reported later in this section.

By comparing with the ground truth, we found the results as shown in Table 5.3. As can been from Table 5.3, FRR in video streams is higher than that in audio streams. This is because the video cameras were placed in such a way that they could not cover the whole corridor, and hence could not detect events outside their coverage area. On the other hand, since the microphones could capture the ambient sound even beyond the corridor area, they were able to detect the events those did not occur in the corridor region. Therefore, the microphones are found to have the FAR higher than that of video streams.

Using our whole set of events, we computed the accuracies (ACC) of event classification for all the four streams. We found the accuracy of individual streams to be moderate. However, it was found that the accuracy of event classification based on video streams (0.60 for both the video streams) was slightly better than that based on audio streams (0.55 for audio stream 1 and 0.51 for audio stream 2). We used these accuracy values to assign the confidences in all the four streams. Note that the overall accuracies of video streams is based on three types of events - “standing”, “walking” and “running”, while the audio streams’ overall accuracies are determined based on five types of events - “walking”, “running”, “talking”, “shouting” and “door knocking”.

Assimilation of all streams

We performed assimilation of the probabilistic decisions obtained from individual streams in four different ways based on whether or not to use the agreement/disagreement information and the confidence information about them. The results are shown in Table 5.4. Note that these results are obtained by setting probability threshold Th and minimum time period t_w to 0.70 and 1 second, respectively.

Overall observations from Table 5.4 are as follows -

- Using multiple streams together provides better overall accuracy (ACC = 0.72) and the reduced False Rejection Rate (FRR = 0.011) as can be seen in the option 1 in Table 5.4. FAR is not evaluated in case of assimilating all the streams; since in the assimilation process, only the evidences of occurrence of the events are used, and therefore it does not affect FAR.

- The results (Table 5.4) imply that using agreement/ disagreement information among the streams is advantageous in obtaining more accurate results, however, using confidence information with it can further improve the overall accuracy of event detection and classification. As can be seen in Table 5.4, option 2, we obtain overall accuracy (ACC = 0.78) by using agreement/disagreement information among streams; which is better compared to the baseline case (ACC = 0.72 in option 1) where the assimilation has been performed using a Bayesian formulation without using the agreement/disagreement and the confidence information . By using confidence information together with the agreement/disagreement information, we obtain the accuracy further improved to (ACC = 0.80), as can be seen at option 4 in Table 5.4.

Table 5.4: Results: Using all the streams with $Th = 0.70$

Option	Agreement coefficient	Confidence information	FRR	ACC
1	No	No	0.011	0.72
2	Yes	No	0.011	0.78
3	No	Yes	0.010	0.76
4	Yes	Yes	0.012	0.80

Note that, the overall accuracies reported in Table 5.4 are for all the events listed in Table 5.1.

Early vs late thresholding

We also observed the accuracy of event classification by varying the probability threshold Th from 0.50 to 0.99. The results are shown in figure 5.12. Figure 5.12 shows how accuracy (ACC) decreases as the probability threshold Th increases for individual streams and for all streams when assimilated with four different options based on whether or not agreement coefficient and confidence information is used.

The observations are as follows -

- It can be clearly seen from figure 5.12 that assimilation of all streams provide better accuracy even with a higher threshold, while individual streams fail in this respect. The accuracy decreases slowly for the combined evidences compared to the individual evidences. This implies that using agreement/disagreement among and confidence information of the streams in the assimilation process not only improves the overall accuracy, it also improves the accuracy of computing the probability of occurrence of the events.

- It also shows that early thresholding of the probabilistic decisions obtained based on individual streams leads to lesser accuracy; for ex-

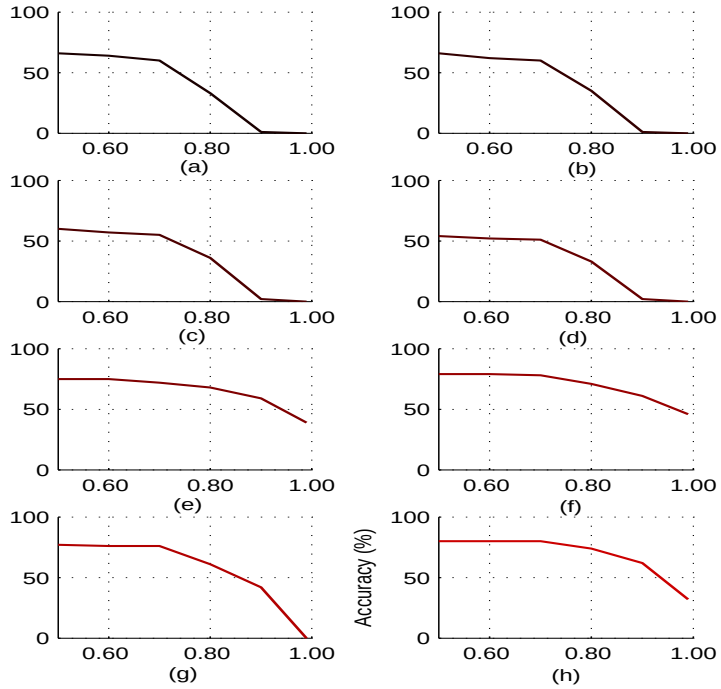

Figure 5.12: Plots: Probability Threshold vs Accuracy. (a) Video stream 1 (b) Video stream 2 (c) Audio stream 1 (d) Audio stream 2 (e)-(h) All streams after assimilation with the four options given in Table 5.4

ample, in figure 5.12, at probability threshold 0.80, we obtain higher accuracies - 68, 71, 61 and 74 in the figures 5.12e-5.12h, respectively, after the assimilation of all streams compared to the accuracies - 33, 35, 36 and 33 in the figures 5.12a-5.12d, respectively, obtained using individual streams.

To summarize the results for information assimilation aspect of our framework, the results have shown that the use of agreement coefficient among and the confidence information of media streams helps in obtaining more accurate and credible decisions about the events. The results have also shown that the False Rejection Rate for event detection can be significantly reduced using all the streams together.

5.3 Optimal Subset Selection Results

This section presents the results for the optimal subset selection of streams in order to detect events in a surveillance scenario [5]. The experiments are performed in the same surveillance setup which has been described in section 5.1. The event $\mathbf{E}_k$ to detect is "A person is walking, knocking the door and then continued walking in the corridor", which has also been as described in Example 3.1 (in section 5.2.4).

To show the utility of our dynamic programming based method for the optimal subset selection of streams, we have considered eight streams which are obtained from four different sensors (two video cameras and two microphones) based on two different sets of features from each of them, as shown in Table 5.5. Note that, in information assimilation results, we used only four streams from four different sensors based on only one feature from each of them. In this case, we have increased the number of streams to eight

because the optimal subset selection of streams makes sense only when the number of streams is significantly large.

Table 5.5(a) shows two different feature sets for the video camera i.e. one set of features is the 'RGB color channel' with 'Blob's displacement', and the second set is the 'RGB color channel' with 'Rate of change in Blob's area'. These features are used for the different classification tasks. How do we use 'Blob's displacement' for detecting video atomic events, has been already described in section 5.2.3. For the feature 'Rate of change in Blob's area', we exploited the fact the blob's area increases at a certain rate as the person moves towards the camera and vica versa.

For audio, as shown in Table 5.5(b), in set 1 of features, we used Zero Crossing Rate (ZCR) feature for all the three classification levels; while in the set 2 of features, we used Root Mean Square (RMS) for foreground/ background segmentation and for distinguishing between the excited and normal events. The Zero Crossing Rate measures the number of times in the given time interval (50 ms in our case) that the signal amplitude passes through a value of zero moving from negative to positive and vice versa. The Root Mean Square is 2-norm of the vector that contains the samples in one audio frame (of 50 ms). The Linear Predictor Coefficients (LPC) are used for categorizing between the vocal and nonvocal events. For the purpose of selecting optimal subset where the cost is also an important constraint, we choose ZCR, RMS and LPC over LFCC because they are relatively less expensive to compute, yet provides decent results.

Preliminary steps of feature extraction, event detection and classification for the video and audio streams are performed as described in section 5.2.3. To assimilate the information obtained from all the eight streams, the probabilistic decisions about the video and audio atomic events are obtained

Table 5.5: The feature used for video and audio streams

(a) Video

Classification task	Set 1 of features	Set 2 of features
Foreground/Background	RGB channels	RGB channels
Running/Walking/Standing	Blob's displacement	Rate of change in Blob's area

(b) Audio

Classification task	Set 1 of features	Set 2 of features2
Foreground/Background	Zero Crossing Rate	Root Mean Square
Vocal/Nonvocal	Zero Crossing Rate	Linear Predictor Coefficients
Excited/Normal	Zero Crossing Rate	Root Mean Square

after every $t_w = 1$ second time (Refer to figure 5.5).

To demonstrate how our dynamic programming based method works, we decompose the compound event $\mathbf{E}_k$ into its constituents atomic events $\mathbf{e}_1$ = "A person walked/stood in the corridor" and $\mathbf{e}_2$ = "A person knocked the door in the corridor". The probabilistic decisions for these two atomic events obtained using 4 video and 4 audio streams are shown along a timeline in figure 5.13. In figure 5.13, x-axis denotes the key points (in steps of seconds) along the timeline and y-axis shows the probability of occurrence of an atomic event based on a particular stream. The legends used are: '$\circ$' - Standing, '$\square$' - Walking and '∇' - Knocking; '$\star$' - No event. For example, the legend '$\circ$' shown at key point '8' for the stream V_{11} indicates the probability of occurrence of an event 'person is standing' based on the feature set 1 (Refer to Table 5.5(a)) obtained from video data of camera 1. We will shortly describe in section 5.3.1 how the optimal subset is selected from the set of these 8 streams.

Cost estimation

As discussed in section 1.1, the cost of using streams usually of two types - one time cost and the running cost. Note that the one time cost (such as installation cost and cost of training classifiers etc) is optimized by the

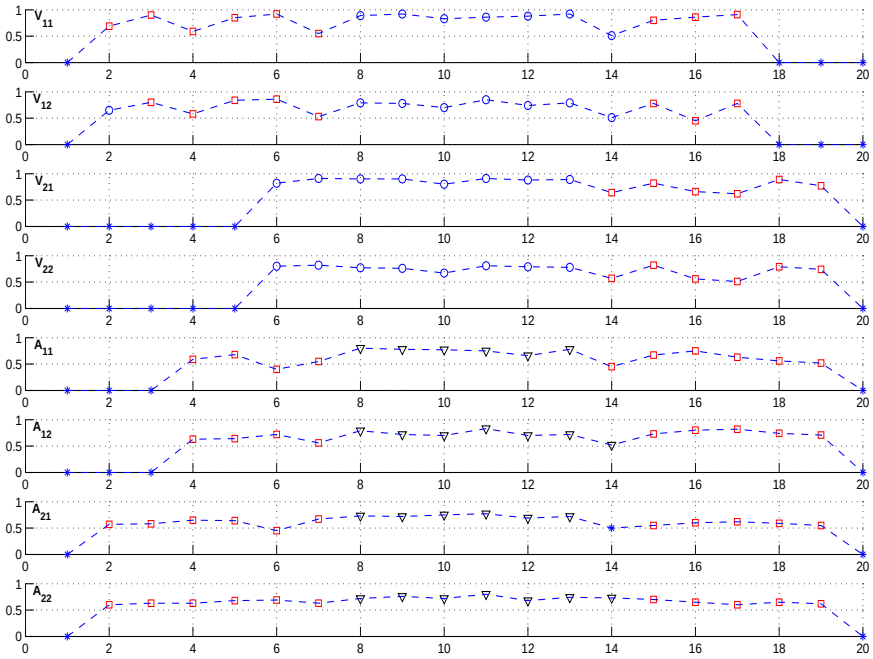

Figure 5.13: Timeline-based probabilistic decisions for the events using all the 8 streams.

system designer during system design. Our focus is on the "on the fly" optimization of running cost by the system. The running cost consists of cost of processing, operating and the wear-tear of the media stream. Note that, the operating and wear-tear cost can be computed based on the statistics of power consumption and diminishing cost of video sensors. For our experiments, we consider only the processing cost of streams and describe how it can be estimated for various video and audio streams.

The processing of stream consists of usually two steps - feature extraction and event classification. We compute the processing cost by estimating the time taken in feature extraction and in event classification steps for all the streams. Table 5.6(a) shows the same for a video stream. For an audio stream, Table 5.6(b) shows the cost of extracting different features (ZCR, RMS and LPC) and the cost of event classification at three different levels.

Table 5.6: The processing cost of video and audio streams

(a) Video stream

Blob detection (BD)	0.66 frames (each of size 756×568) per second
Event classification (EC)	0.010 seconds

(Assuming that there are 8 frames per second in video, it takes 8/0.66 ≈ 12.12 seconds for processing of 1 second of video)

(b) Audio stream

Feature extraction	ZCR	RMS	LPC
Cost	1.5642 seconds	0.8628 seconds	1.5072 seconds
Event classification	Foreground/Background (F/B)	Excited/Normal (E/N)	Vocal/Nonvocal (V/NV)
Cost	0.0082 seconds	0.0076 seconds	0.0100 seconds

(These processing costs are for 1 second of audio)

(c) The total estimated cost for all the streams

Stream	Cost breakup	Estimated total cost (in Unit money)
V_{11}, V_{12}, V_{21}, V_{22}	(12.12 (BD) + 0.010 (EC))× 1	≈ 12.0
A_{11}, A_{21}	(1.5642 (ZCR)+ 0.0082 (F/B) + 0.0076 (E/N) + 0.0100 (V/NV))× 1	≈ 1.5
A_{12}, A_{22}	(0.8628 (RMS)+ 1.5072 (LPC) + 0.0082 (F/B) + 0.0076 (E/N) + 0.0100 (V/NV))× 1	≈ 2.5

(These costs are for processing of streams of 1 second. In calculating the final cost, we assume that the processing of every second of data costs 1 unit money)

Based on the data shown in Table 5.6(a) and Table 5.6(b), we provide the total estimated cost for all the 8 streams in Table 5.6(c). Note that when the two video streams obtained from the same camera (e.g. V_{11},V_{12} from camera 1 or V_{21},V_{22} from camera 2) are together selected in the optimal subset, the cost of only one stream is counted since the major cost of blob detection remains common in both.

Computing confidences in streams

We computed the confidences in all the four video streams used by running the experiments for the data set given in Table 5.1. By comparing results with the ground truth, we noticed that the event detection was found 60% times correct using the feature sets 1 (i.e. RGB color channel and blob's displacement) of both the camera 1 and camera 2; while it was found 55%

Table 5.7: The confidences in all the streams

Stream	V_{11}	V_{12}	V_{21}	V_{22}	A_{11}	A_{12}	A_{21}	A_{22}
Confidence	0.60	0.55	0.60	0.54	0.55	0.58	0.55	0.58

and 54%, respectively, with feature set 2 (i.e RGB color channel and blob's area). The audio analysis was done separately [7] and it was found that the overall accuracy of event detection using audio sensors was 55% based on ZCR and was 58% based on (RMS+LPC). Based on this experimental evidence, we assigned the confidence levels to different streams as shown in Table 5.7.

5.3.1 Optimal subset selection of streams

In this section, we show how our framework selects the optimal subset of streams for detecting the event $\mathbf{E}_k$. Note that, due to the placement and the coverage space of sensors, all the sensors may not detect the event at the same time instance. Therefore, the environment information is needed to determine the right set of streams out of which optimal subset would be selected. As shown in figure 5.13, the event **E** is detected based on the set (V_{11}, V_{12}, A_{21}, A_{22}) of streams at key point '2'.

In the subsequent paragraphs, we first show how the optimal subset is computed at a key point. Next, we demonstrate how frequently the optimal subset is recomputed along the timeline and also how much cost is saved by using only the optimal subset.

Finding optimal subset at a key-point

The system computes the optimal subset at key point '2' as follows. First, since the probabilistic decisions based on the three (V_{11}, A_{21}, A_{22}) of four

streams are in favor of the "walking" event, they are kept into group S_1 and the rest (V_{12}) is kept into group S_2 (Refer to section 3.4.2, Step 3). Next, we assimilate the probabilistic decisions obtained based on the streams within each of the two sets and obtain the fused probabilities $P(\mathbf{E}_k|S_1)$ and $P(\bar{\mathbf{E}}_k|S_2)$ using equation (3.8) by assuming an uniform agreement coefficient $\gamma = 0$ among the streams. Note that we have described in section 5.2.5 how the agreement or disagreement among the streams affects fused probabilities. We also find the overall confidence F_{S_1} and F_{S_2} of the two sets S_1 and S_2, respectively, using equation (3.13). We obtain $P(\mathbf{E}_k|S_1) = 0.82$, $P(\bar{\mathbf{E}}_k|S_2) = 0.65$, $F_{S_1} = 0.72$ and $F_{S_2} = 0.55$. Since $P(\mathbf{E}_k|S_1).F_{S_1} = 0.5904) > (P(\bar{\mathbf{E}}_k|S_2).F_{S_2} = 0.3575)$, we conclude that there is more evidence in support of the "walking" event compared to the evidences in favor of the "standing" event.

The optimal subset is then found from set S_1 using a dynamic programming based framework described in Section 4.4.1 (**MaxGoal** - for maximizing probability), Section 4.4.2 (**MaxConf** - for maximizing confidence) and Section 4.4.3 (**MinCost** - for minimizing cost). The optimal subset process at key point '2' is depicted in figure 5.14. Figure 5.14a plot shows how probability is maximized under the given cost constraints, and figure 5.14b depicts how confidence varies with respect to cost as a result of maximizing the probability, using the subsets denoted by symbols **A**, **B** etc. A similar explanation holds true for figure 5.14c-5.14f.

The overall observations from the figure 5.14a-5.14f are:

1. The proposed framework allows for a tradeoff among the extent to which the goal is achieved, the confidence with which the goal is achieved and the cost of achieving the goal. It offers the flexibility to compare whether any one set of streams of low cost would be better

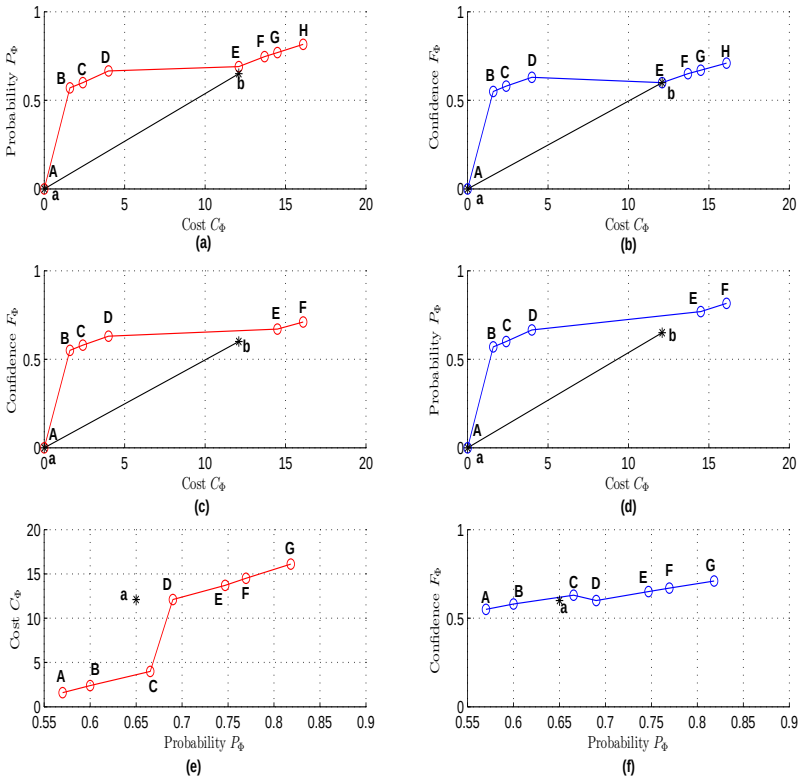

Figure 5.14: (a) and (b) **MaxGoal**: **A** = (Nil), **B** = (A_{21}), **C** = (A_{22}), **D** = (A_{21}, A_{22}), **E** = (V_{11}), **F** = (V_{11}, A_{21}), **G** = (V_{11}, A_{22}), **H** = (V_{11}, A_{21}, A_{22}) represent the subsets in favor of event "walking"; (c) and (d) **MaxConf**: **A** to **D** - Same as **MaxGoal**, **E** = (V_{11}, A_{22}), **F** = (V_{11}, A_{21}, A_{22}) represent the subsets in favor of event "walking"; (e) and (f) **MinCost**: **A**= (A_{21}), **B**= (A_{22}), **C** = (A_{21}, A_{22}), **D** = (V_{11}), **E** = (V_{11}, A_{21}), **F** = (V_{11}, A_{22}), **G** = (V_{11}, A_{21}, A_{22}) represent the subsets in favor of event "walking"; and the symbols **a** = (Nil), **b** = (A_{12}) represent the subsets in favor of event "standing" for all three MS problems.

than any other set of streams of higher cost, or any one set of media streams of high confidence would be better than any other set of streams low confidence. For instance, figure 5.14a clearly shows that the subset indicated by symbol **D** would be better to choose than the subset indicated by symbol **E** since there is a very small difference in the goal achieved (and in the overall confidence) using the two subsets (**D** helps in detecting the event with 0.03 less probability than **E** and with overall confidence more than that in **E**) while there is a significant difference (of ≈ 8) in the cost.

2. The framework also allows for a tradeoff - whether one should opt for maximizing probability, for maximizing confidence or for minimizing cost. The plots in figure 5.14 suggest how the second factor (say probability of occurrence of event) varies with the third factor (say cost) if one opts for maximizing the first factor (say confidence). The same also holds true for other combinations.

3. The graphs (in figure 5.14) show a pictorial representation of which subset of streams is most suitable in terms of optimal probability, optimal confidence or the optimal cost. It also helps in deciding which is the next most suitable subset in case the best subset is not available. For instance, in figure 5.14e, let subset denoted by **G** is in use. If at some instant the stream A_{21} is unavailable, we can find from the plot that the next best subset is the one denoted by **F**.

Finding optimal subset along a timeline

Once the optimal subset is computed at key point '2', the system continues using this subset along the timeline while ignoring the other streams until

the probability of occurrence of event using this subset does not fall below a threshold (0.80, in our experiment). If probability value falls below threshold, the optimal subset is *recomputed* using *all* the available streams. The processing cost of the streams which are ignored is saved.

The timeline-based statistics of subset used for detecting the event $\mathbf{E}_k$, the loss in probability P_Φ of occurrence of event and in confidence F_Φ in the subset used, and the savings in cost C_Φ (of processing the subset) using all the three methods **MaxGoal**, **MaxConf** and **MinCost** are provided in Table 5.8, Table 5.9 and Table 5.10, respectively. Note that the cost of processing the full set (i.e. all the eight streams) is 32, the maximum overall probability of occurrence of event is 0.99 and the maximum overall confidence is 0.90 when the full set of streams is used.

The key observations from the Table 5.8 to Table 5.10 are as follows:

1. The proposed framework for selecting the optimal subset selection along a timeline provides significant savings in processing cost at the marginal loss in the overall probability of achieved goal and in the overall confidence in the subset used. As can be seen from Tables 5.8-5.10, the savings in cost C_Φ of 10.2 unit ($\approx$ 32% for **MaxGoal**), 7.4 unit ($\approx$ 23% for **MaxConf**) and 16.8 unit ($\approx$ 50% for **MinCost**) per key point (which occur at every second) is achieved at the expense of approximately 5% and 15% loss in probability P_Φ and confidence F_Φ, respectively.

2. The method **MinCost**, although provides better savings in cost but fails to detect a few atomic events at some key points. For instance, the method in an effort to minimize the cost selects only the audio streams in the optimal subset which could detect only "door knocking" atomic event; but in absence of video streams, it fails to detect whether the

Table 5.8: Timeline-based optimal subset selection using **MaxGoal**

Key point	**Description**	**Loss in P_Φ**	**Loss in F_Φ**	**Saving in C_Φ**
1	No event	-	-	-
2	**All the available streams used and the optimal subset Φ computed** Walk: Φ = (V_{11}, A_{21}, A_{22}), $P_\Phi = 0.95$, $F_\Phi = 0.72$, $C_\Phi = 16$	0	0	0
3	Φ used: (V_{11}, A_{21}, A_{22}), Walk: $P_\Phi = 0.95$, $F_\Phi = 0.72$, $C_\Phi = 16$	0.04	0.18	16
4	Φ used: (V_{11}, A_{21}, A_{22}), Walk: $P_\Phi = 0.77$, $F_\Phi = 0.72$, $C_\Phi = 16$ Since $P_\Phi < P_{spec} \Rightarrow$ **Optimal subset Φ recomputed**, Walk: Φ = (V_{11}, A_{11}, A_{12}, A_{21}, A_{22}), $P_\Phi = 0.89$, $F_\Phi = 0.81$, $C_\Phi = 20$	0	0	0
5	Φ used: (V_{11}, A_{11}, A_{12}, A_{21}, A_{22}), Walk: $P_\Phi = 0.99$, $F_\Phi = 0.81$, $C_\Phi = 20$	0	0.09	12
6	Φ used: (V_{11}, A_{11}, A_{12}, A_{21}, A_{22}), Walk (V_{11}, A_{12}, A_{22}): $P_\Phi = 0.99$, $F_\Phi = 0.74$, $C_\Phi = 17$ Stand (A_{11}, A_{21}): $P_\Phi = 0.73$, $F_\Phi = 0.60$, $C_\Phi = 3$	0	0.16	12
7	Φ used: (V_{11}, A_{11}, A_{12}, A_{21}, A_{22}) Walk: $P_\Phi = 0.87$, $F_\Phi = 0.81$, $C_\Phi = 20$	0.12	0.09	12
8	Φ used: (V_{11}, A_{11}, A_{12}, A_{21}, A_{22}), Stand (V_{11}): $P_\Phi = 0.89$, $F_\Phi = 0.60$, $C_\Phi = 12$ Knock (A_{11}, A_{12}, A_{21}, A_{22}): $P_\Phi = 0.99$, $F_\Phi = 0.74$, $C_\Phi = 8$	0.10	0.30	12
9	Φ used: (V_{11}, A_{11}, A_{12}, A_{21}, A_{22}), Stand (V_{11}): $P_\Phi = 0.92$, $F_\Phi = 0.60$, $C_\Phi = 12$ Knock (A_{11}, A_{12}, A_{21}, A_{22}): $P_\Phi = 0.99$, $F_\Phi = 0.74$, $C_\Phi = 8$	0.07	0.30	12
10	Φ used: (V_{11}, A_{11}, A_{12}, A_{21}, A_{22}), Stand (V_{11}): $P_\Phi = 0.83$, $F_\Phi = 0.60$, $C_\Phi = 12$ Knock (A_{11}, A_{12}, A_{21}, A_{22}): $P_\Phi = 0.98$, $F_\Phi = 0.74$, $C_\Phi = 8$	0.16	0.30	12
11	Φ used: (V_{11}, A_{11}, A_{12}, A_{21}, A_{22}), Stand (V_{11}): $P_\Phi = 0.86$, $F_\Phi = 0.60$, $C_\Phi = 12$ Knock (A_{11}, A_{12}, A_{21}, A_{22}): $P_\Phi = 0.99$, $F_\Phi = 0.74$, $C_\Phi = 8$	0.13	0.30	12
12	Φ used: (V_{11}, A_{11}, A_{12}, A_{21}, A_{22}), Stand (V_{11}): $P_\Phi = 0.88$, $F_\Phi = 0.60$, $C_\Phi = 12$ Knock (A_{11}, A_{12}, A_{21}, A_{22}): $P_\Phi = 0.96$, $F_\Phi = 0.74$, $C_\Phi = 8$	0.11	0.30	12
13	Φ used: (V_{11}, A_{11}, A_{12}, A_{21}, A_{22}), Stand (V_{11}): $P_\Phi = 0.92$, $F_\Phi = 0.60$, $C_\Phi = 12$ Knock (A_{11}, A_{12}, A_{21}, A_{22}): $P_\Phi = 0.99$, $F_\Phi = 0.74$, $C_\Phi = 8$	0.07	0.30	12
14	Φ used: (V_{11}, A_{11}, A_{12}, A_{21}, A_{22}), Walk (V_{11}): $P_\Phi = 0.51$, Since $P_\Phi < P_{spec} \Rightarrow$ **Optimal subset Φ recomputed**, Walk: Φ = (V_{21}, V_{22}, A_{11}), $P_\Phi = 0.74$, $F_\Phi = 0.68$, $C_\Phi = 13.5$ Stand (V_{11}, V_{12}): $P_\Phi = 0.52$, $F_\Phi = 0.65$, $C_\Phi = 12$ Knock (A_{12}, A_{22}): $P_\Phi = 0.75$, $F_\Phi = 0.66$, $C_\Phi = 5$	0	0	0
15	Since $P_\Phi < P_{spec}$ at point 14 $\Rightarrow$ **Optimal subset Φ recomputed**, Walk: Φ = (V_{21}, V_{22}, A_{11}, A_{12}), $P_\Phi = 0.99$, $F_\Phi = 0.75$, $C_\Phi = 16$	0	0	0
16	Φ used: (V_{21}, V_{22}, A_{11}, A_{12}), Walk: $P_\Phi = 0.99$, $F_\Phi = 0.75$, $C_\Phi = 16$	0	0.15	16
17	Same as key point 16	0	0.15	16
18	Φ used: (V_{21}, V_{22}, A_{11}, A_{12}), Walk: $P_\Phi = 0.93$, $F_\Phi = 0.67$, $C_\Phi = 16$, No event (V_{21})	0.06	0.23	16
19	Φ used: (V_{22}, A_{11}, A_{12}), Walk: $P_\Phi = 0.88$, $F_\Phi = 0.67$, $C_\Phi = 16$	0.11	0.23	16
20	Φ used: (V_{22}, A_{11}, A_{12}), No event,$C_\Phi = 16$	0	0	16
	Average losses and savings per key point	**0.0485**	**0.154**	**10.2**

person is standing, walking or running.

3. Since the processing cost of the optimal subset is significantly reduced compared to the cost of the full set of streams, it helps in achieving the real-time performance in the event detection.

Table 5.9: Timeline-based optimal subset selection using **MaxConf**

Key point	**Description**	**Loss in P_Φ**	**Loss in F_Φ**	**Saving in C_Φ**
1-14	Same as Table 5.8			
15	Since $P_\Phi < P_{spec}$ at point 14 ⇒ **Optimal subset Φ recomputed**, Walk: Φ = (V_{11}, V_{21}, A_{11}, A_{12}), $P_\Phi = 0.99$, $F_\Phi = 0.79$, $C_\Phi = 28$	0	0	0
16	Walk: Φ = (V_{11}, V_{21}, A_{11}, A_{12}), $P_\Phi = 0.99$, $F_\Phi = 0.79$, $C_\Phi = 28$	0	0.21	4
17	Same as key point 16	0	0.21	4
18	Φ used: (V_{11}, V_{21}, A_{11}, A_{12}), Walk (A_{11}, A_{12}): $P_\Phi = 0.78$, $F_\Phi = 0.63$, $C_\Phi = 4$, No event (V_{11}, V_{21}): $C_\Phi = 24$	0.21	0.27	4
19	Since $P_\Phi < P_{spec}$ at point 18 ⇒ **Optimal subset Φ recomputed**, Walk: Φ = (V_{21}, A_{11}, A_{12}, A_{21}, A_{22}), $P_\Phi = 0.95$, $F_\Phi = 0.81$, $C_\Phi = 20$	0	0	0
20	Φ used: (V_{21}, A_{11}, A_{12}, A_{21}, A_{22}), No event, $C_\Phi = 20$	0	0	12
	Average losses and savings per key point	**0.0505**	**0.151**	**7.4**

Table 5.10: Timeline-based optimal subset selection using **MinCost**

Key point	**Description**	**Loss in P_Φ**	**Loss in F_Φ**	**Saving in C_Φ**
1	No event	-	-	-
2	**All the available streams used and the optimal subset Φ computed** Walk: Φ = (V_{11}, A_{21}, A_{22}), $P_\Phi = 0.95$, $F_\Phi = 0.72$, $C_\Phi = 16$	0	0	0
3	Φ used: (V_{11}, A_{21}, A_{22}), Walk: $P_\Phi = 0.95$, $F_\Phi = 0.72$, $C_\Phi = 16$	0.04	0.18	16
4	Φ used: (V_{11}, A_{21}, A_{22}), Walk: $P_\Phi = 0.77$, $F_\Phi = 0.72$, $C_\Phi = 16$ Since $P_\Phi < P_{spec}$ ⇒ **Optimal subset Φ recomputed**, Walk: Φ = (A_{11}, A_{12}, A_{22}), $P_\Phi = 0.81$, $F_\Phi = 0.70$, $C_\Phi = 6.5$	0	0	0
5	Φ used: (A_{11}, A_{12}, A_{22}), Walk: $P_\Phi = 0.81$, $F_\Phi = 0.70$, $C_\Phi = 6.5$	0.18	0.20	25.5
6	Φ used: (A_{11}, A_{12}, A_{22}), Walk (A_{12}, A_{22}): $P_\Phi = 0.85$, $F_\Phi = 0.66$, $C_\Phi = 5$ Stand (A_{11}): $P_\Phi = 0.69$, $F_\Phi = 0.55$, $C_\Phi = 1.5$	0.14	0.24	25.5
7	Φ used: (A_{11}, A_{12}, A_{22}), Walk: $P_\Phi = 0.73$, $F_\Phi = 0.70$, $C_\Phi = 6.5$ Since $P_\Phi < P_{spec}$ ⇒ **Optimal subset Φ recomputed**, Walk: Φ = (A_{11}, A_{21}, A_{22}), $P_\Phi = 0.81$, $F_\Phi = 0.67$, $C_\Phi = 5.5$	0	0	0
8	Φ used: (A_{11}, A_{21}, A_{22}), Knock: $P_\Phi = 0.97$, $F_\Phi = 0.67$, $C_\Phi = 5.5$	0.02	0.23	26.5
9	Same as key point 8	0.02	0.23	26.5
10	Same as key point 8 except $P_\Phi = 0.96$	0.03	0.23	26.5
11	Same as key point 8 except $P_\Phi = 0.98$	0.01	0.23	26.5
12	Same as key point 8 except $P_\Phi = 0.90$	0.09	0.23	26.5
13	Same as key point 8 except $P_\Phi = 0.96$	0.03	0.23	26.5
14	Φ used: (A_{11}, A_{21}, A_{22}), Knock (A_{22}): $P_\Phi = 0.73$, Since $P_\Phi < P_{spec}$ ⇒ **Optimal subset Φ recomputed**, Knock (A_{11}, A_{12}): $P_\Phi = 0.85$, $F_\Phi = 0.63$, $C_\Phi = 4$	0	0	0
15	Knock (A_{11}, A_{12}): $P_\Phi = 0.85$, $F_\Phi = 0.63$, $C_\Phi = 4$	0.14	0.27	28
16	Same as key point 15 except $P_\Phi = 0.92$	0.07	0.27	28
17	Same as key point 15 except $P_\Phi = 0.89$	0.10	0.27	28
18	Same as key point 15 except $P_\Phi = 0.78$ Since $P_\Phi < P_{spec}$ ⇒ **Optimal subset Φ recomputed**, Walk (A_{12}, A_{21}): $P_\Phi = 0.80$, $F_\Phi = 0.63$, $C_\Phi = 4$	0	0	0
19	Φ used: (A_{12}, A_{21}), Walk: $P_\Phi = 0.75$ Since $P_\Phi < P_{spec}$ ⇒ **Optimal subset Φ recomputed**, Walk (A_{12}, A_{21}, A_{22}): $P_\Phi = 0.83$, $F_\Phi = 0.70$, $C_\Phi = 6.5$	0	0	0
20	Φ used: (A_{12}, A_{21}, A_{22}), No event, $C_\Phi = 6.5$	0	0	25.5
	Average losses and savings per key point	**0.0415**	**0.141**	**16.8**

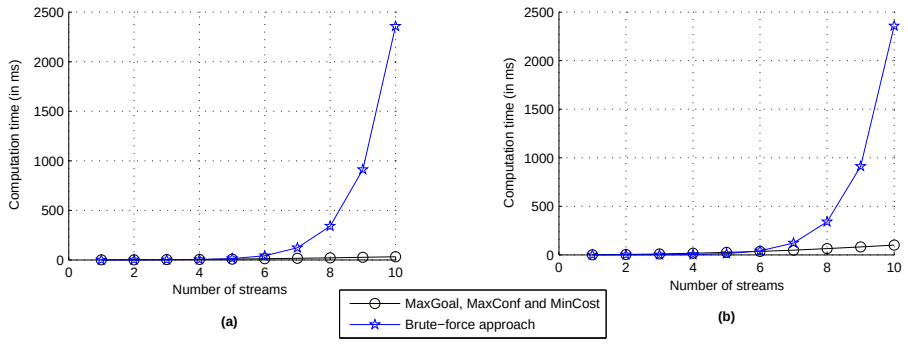

Figure 5.15: Comparison of (a) **MaxGoal** and **MaxConf** (with $C_n = 32$), (b) **MinCost** (with $L = 100$), with the brute-force approach

The proposed method versus the brute-force approach

We have compared our dynamic programming based method for stream subset selection with the brute force approach by recording the computation time for varying number of streams, as shown in figure 5.15. In **MaxGoal** and **MaxConf**, the total cost is taken as 32; and in **MinCost**, the total number of discrete levels L of probability values is taken as 100. The plots in figure 5.15, show that the computation time taken by the dynamic programming based method is significantly lesser compared to the brute-force approach as the number of streams increases.

5.4 Results Summary

In this chapter, we have presented the experimental results from two different perspectives - first, 'when' and 'how' to assimilate the information obtained from various sources; and second, 'what' information to assimilate i.e. how to find the optimal subset of streams which should be assimilated to accomplish a task subject to the specified constraints.

For the first, in section 5.2, we have shown how the proposed framework

integrates the agreement/disagreement coefficient among and the confidence information of streams in combining them for detecting events in a surveillance scenario. The experimental results have shown that the use of agreement coefficient among and the confidence information of media streams helps in obtaining more accurate and credible decisions about the events. The results have also shown that the False Rejection Rate for event detection can be significantly reduced using all the streams together.

For the second perspective, in section 5.3, we have shown through experiments that the proposed framework allows for a tradeoff among the three above-mentioned criteria, and offers a flexibility to compare whether any one set of media streams of low cost would be better than any other set of media streams of higher cost, or any one set of media streams of high confidence would be better than any other set of media streams low confidence. The experimental results have shown the utility of the dynamic programming based method for detecting events in a surveillance scenario. The results have shown that the subset of a significantly lower cost can help in detecting events at the expense of minor loss in the probability and the confidence with which the events are detected.

Chapter 6

Conclusions and Future Research Directions

This dissertation has presented a novel framework for assimilation of information in order to detect events in the surveillance and monitoring systems that utilize multifarious sensors. The framework has addressed the issues of 'when' to assimilate the information, 'how' to assimilate the information and 'what' information to assimilate for better detection of atomic and compound events in a multimedia surveillance environment. The solutions to first two issues 'when' and 'how' to assimilate the information has been described in Chapter 3, and the issue of 'what' information to assimilate i.e. to determine the optimal subset of streams has been addressed in Chapter 4.

In Chapter 3, we have presented hierarchical probabilistic assimilation approach for detecting compound/atomic events. It is shown how assimilation takes place at three different levels - media stream level, atomic event level, and compound event level. A method for computing the agreement coefficient between any two streams is described. The fusion models for the

agreement coefficient and for the confidence information are also presented in this chapter. The corresponding experimental results (in section 5.2 of Chapter 5) have demonstrated that the use of agreement coefficient among streams and the confidence information of media streams helps in obtaining more accurate and credible decisions about the events. The results have also shown that the False Rejection Rate for event detection can be significantly reduced using all the streams together.

Chapter 4 have described a dynamic programming approach to determine the optimal subset of media streams for three different objectives - maximizing the probability of achieving the goal under the specified cost and confidence constraints; maximizing the confidence in the achieved goal under the specified cost and probability constraints; and minimizing the cost of using the subset to obtain a specified probability of achieving the goal with a specified confidence. Each of these problems is proven to be NP-Complete, after which we have proposed a dynamic programming approach that finds the optimal subset of media streams based on the above three criteria. From an AI point of view, the solution we propose, is heuristic-based, and for each criterion, it utilizes a heuristic function which, for a given problem, combines *optimal* solutions of small-sized sub-problems to yield a potential near-optimal solution to the original problem. The corresponding experimental results (in section 5.3 of Chapter 5) have established the utility of the framework for detecting events in a surveillance scenario. The results have shown that the subset of a significantly lower cost can help in detecting events at the expense of minor loss in the probability and the confidence with which the goal is achieved.

6.1 Conclusions

Based on the work presented in this thesis, we can draw the following conclusions -

1. The late assimilation strategy is advantageous over early assimilation since it offers scalability (i.e. graceful upgradation or degradation) in terms of media streams used in the assimilation process.

2. Use of agreement/disagreement among the streams and the confidence in each stream in the assimilation process helps in improving the overall accuracy of event detection in multimedia surveillance systems.

3. The Media Selection problems introduced in this thesis are NP-Complete.

4. Though the three Media Selection problems for selecting the optimal subset of streams are NP-Complete, the dynamic programming based approach finds the optimal subset of media streams in *pseudo*-polynomial time.

5. The dynamic programming based method allows for a tradeoff among the three criteria - maximizing the probability of achieving the goal under the specified cost and confidence constraints; maximizing the confidence in the achieved goal under the specified cost and probability constraints; and minimizing the cost of using the subset to obtain a specified probability of achieving the goal with a specified confidence.

6. The proposed approach offers the flexibility to compare whether any one set of media streams of low cost would be better than any other set of media streams of higher cost, or any one set of media streams of high confidence would be better than any other set of media streams of lower confidence.

7. The proposed approach also offers the user a flexibility to choose alternative (or the next best) subsets when the best subset is unavailable.

6.2 Future Research Directions

This dissertation proposes a novel information assimilation framework that exposes several direction of research. This thesis has used a fixed-time-interval based strategy (in Chapter 3) to determine 'when' the information obtained from different sources should be assimilated, however, there are many other related issues which need to explored such as - first, how to determine the minimum time period to confirm different events; second, it would be interesting to see how the framework will work when the information from different sources would be made available at different time instances, what would be the ideal sampling rate of event detection and information assimilation; and finally, how the confidence information about a stream (newly added in the system) can be computed over time using its agreement/disagreement with the other streams whose confidence information are known, and how it would evolve over time with the changes in environment. We have shown the utility of the proposed information assimilation framework in a surveillance scenario, however, it would be interesting to explore how the framework can be customized for other applications such as media-search (or event-search) etc.

The dynamic programming based approach for optimal subset selection of streams proposed in Chapter 4 opens up several research questions. It would be interesting to see how the proposed approach can be used in other scenarios such as for selecting streams in media search systems, and for selecting an optimal subset of streams from a media-server for play or for transmitting onto a network. There is also a need to focus on the for-

malization of how frequently the approximately-optimal subset should be re-computed. Although the method proposed in thesis has focused on multimedia inputs, it would also interesting to foresee a similar problem with respect to multimedia output where one would try to determine the minimal subset of multimedia streams to communicate an intent.

6.2.1 Broad vision: Surveillance in a "search paradigm"

Current surveillance systems, which cost significant amounts of money, are usually designed to handle only the specified task(s) in a rigid sensor settings. For example, if a surveillance system is designed to capture the faces of persons entering into a designated area, it is hardly used for performing any other task.

We prefer to adopt a flexible approach and look at the surveillance systems in a "search paradigm" where an end-user queries the system, in a continuous or one-time manner, for the events of interest. Our vision for multimedia surveillance systems advocates for end-user to have flexibility of defining domain-events at run-time using the data-events and the environment information. This is in contrary to the hardwiring of events at the compile-time.

The proposed system would have many challenging research issues [42]. Some of them are identified as Information assimilation, Domain-data transformation modeling, and Environment modeling.

Information assimilation

Information assimilation involves issues of combining information obtained from multiple heterogeneous sensors. This dissertation has focused on the issue of information assimilation. However, other issues remain to be explore

in future research. We briefly discuss below the other two issues.

Domain-data transformation modeling

Domain-data transformation modeling involves research issues of how to develop a model which can transform a domain-event query to data-event query at run-time. It would be interesting to explore whether rule-based mapping or the script language programming can be used to develop such a model. To incorporate a new query by the user, how to update the model is also another scalability issue.

Environment modeling

Environment modeling requires a model that describes an environment in a generic and scalable manner. Given a location in the environment under surveillance, the system should be able to identify the sensors and other sources that can be used to detect specified events in that environment. In addition, adding/removing of sensors from the environment (scalability) would also be handled.

Bibliography

[1] P. K. Atrey and M. S. Kankanhalli. Probability fusion for correlated multimedia streams. In *ACM International Conference on Multimedia*, pages 408–411, NY, USA, October 2004.

[2] P. K. Atrey and M. S. Kankanhalli. Goal based optimal selection of media streams. In *IEEE International Conference on Multimedia and Expo*, pages 305–308, Amsterdam, The Netherlands, July 2005.

[3] P. K. Atrey, M. S. Kankanhalli, and R. Jain. Timeline-based information assimilation in multimedia surveillance and monitoring systems. In *The ACM International Workshop on Video Surveillance and Sensor Networks*, pages 103–112, Singapore, November 2005.

[4] P. K. Atrey, M. S. Kankanhalli, and R. Jain. Information assimilation framework for event detection in multimedia surveillance systems. *Special Issue on Multimedia Surveillance Systems in Springer/ACM Multimedia Systems Journal*, 12(3):239–253, December 2006.

[5] P. K. Atrey, M. S. Kankanhalli, and J. B. Oommen. Goal-oriented optimal subset selection of correlated multimedia streams. *ACM Transactions on Multimedia Computing, Communications and Applications*, 2007. (Accepted, To appear).

[6] P. K. Atrey, V. Kumar, A. Kumar, and M. S. Kankanhalli. Experiential sampling based foreground/background segmentation for video surveillance. In *IEEE International Conference on Multimedia and Expo*, Toronto, Canada, July 2006.

[7] P. K. Atrey, N. C. Maddage, and M. S. Kankanhalli. Audio based event detection for multimedia surveillance. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages V813–816, Toulouse, France, May 2006.

[8] N. Babaguchi, Y. Kawai, and T. Kitahashi. Event based indexing of broadcasted sports video by intermodal collaboration. *IEEE Transactions on Multimedia*, 4:68–75, March 2002.

[9] N. Babaguchi and N. Nitta. Intermodal collaboration: A strategy for semantic content analysis for broadcast sports video. In *IEEE International Conference on Image Processing*, 2003.

[10] M. J. Beal, N. Jojic, and H. Attias. A graphical model for audio-visual object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25:828– 836, July 2003.

[11] J. A. Benediktsson and I. Kanellopoulos. Classification of multisource and hyperspectral data based on decision fusion. *IEEE Trans. on Geo-Science and Remote Sensing*, 37(3):1367–1377, May 1999.

[12] S. Bhonsle, A. Gupta, S. Santini, M. Worring, and R. Jain. Complex visual activity recognition using a temporal ordered database. In *International Conference on Visual Information Management*, pages 719–726, Amsterdam, The Netherlands, June 1999.

[13] D. A. Bloch and H. C. Kraemer. 2 × 2 Kappa coefficients: Measures of agreement or association. *Journal of Biometrics*, 45(1):269–287, 1989.

[14] F. Brmond and M. Thonnat. A context representation of surveillance systems. In *European Conference on Computer Vision*, Orlando, Florida, May 1996.

[15] R. R. Brooks and S. S. Iyengar. *Multi-sensor fusion: Fundamentals and applications with software*. Upper Saddle River, N.J. : Prentice Hall PTR, 1998.

[16] J. E. S. Cande, A. Teuner, S. B. Park, and B. J. Hosticka. Surveillance system based on detection and tracking of moving objects using CMOS imagers. In *IEEE International Conference Computer Vision Systems*, pages 432–449, Las Palmas, Gran Canaria, Spain, January 1999.

[17] Z. Chair and P. R. Varshney. Optimal data fusion in multiple sensor detection systems. *IEEE Transactions on Aerospace and Electronic Systems*, 22:98–101, 1986.

[18] L. Chaisorn, T.-S. Chua, C.-H. Lee, Y. Zhao, H. Xu, H. Feng, and Q. Tian. A multi-modal approach to story segmentation for news video. 6:187–208, June 2003.

[19] E. Chang and Y. F. Wang. Multi-camera spatio-temporal fusion and biased sequence-data learning for security surveillance. In *ACM International Workshop on Video Surveillance*, Berkley, CA, USA, November 2003.

[20] N. Checka, K. W. Wilson, M. R. Siracusa, and T. Darrell. Multiple person and speaker activity tracking with a particle filter. In *International Conference on Acoustics Speech and Signal Processing*, 2004.

[21] J. Chen and N. Ansari. Adaptive fusion of correlated local decisions. *IEEE Trans. on Systems, Man, and Cybernetics*, 28(2):276–281, May 1998.

[22] H. L. Chieu and Y. K. Lee. Query based event extraction along a timeline. In *International ACM SIGIR Conference on Research and development in Information Retrieval*, pages 425–432, Sheffield, UK, July 2004.

[23] C. Clavel, T. Ehrette, and G. Richard. Event detection for an audio-based surveillance system. In *IEEE International Conference on Multimedia and Expo*, Amsterdam, July 2005.

[24] R. T. Collins, A. Lipton, H. Fujiyoshi, and T. Kanade. Algorithms for cooperative multisensor surveillance. *Proceedings of IEEE*, 89(10):1456–1477, 2001.

[25] M. Cristani, M. Bicego, and V. Murino. Online adaptive background modeling for audio surveillance. In *IEEE International Conference on Pattern Recognition*, pages 399–402, Cambridge, UK, August 2004.

[26] R. Debouk, S. Lafortune, and D. Teneketzis. On an optimal problem in sensor selection. *Journal of Discrete Event Dynamic Systems: Theory and Applications*, 12:417–445, 2002.

[27] A. Dufaux, L. Bezacier, M. Ansorge, and F. Pellandini. Automatic sound detection and recognition for noisy environment. In *European Signal Processing Conference*, pages 1033–1036, Finland, September 2000.

[28] H. Feng, R. Shi, and T.-S. Chua. A bootstrapping framework for annotating and retrieving WWW images. In *ACM International Conference on Multimedia*, pages 960–967, New York City, NY, USA, October 2004.

[29] J. Fisher-III, T. Darrell, W. Freeman, and P. Viola. Learning joint statistical models for audio-visual fusion and segregation. In *Advances in Neural Information Processing Systems*, pages 772–778, Denver, Colorado, November 2000.

[30] G. L. Foresti and L. Snidaro. A distributed sensor network for video surveillance of outdoor environments. In *IEEE International Conference on Image Processing*, Rochester, New York, USA, September 2002.

[31] M. Gandetto, L. Marchesotti, S. Sciutto, D. Negroni, and C. S. Regazzoni. Multi-sensor surveillance towards smart interactive spaces. In *IEEE International Conference on Multimedia and Expo*, pages I:641–644, Baltimore, MD, USA, July 2003.

[32] C. Genest and J. V. Zidek. Combining probability distributions: A critique and annotated bibliography. *Journal of Statistical Science*, 1(1):114–118, 1986.

[33] D. L. Hall and J. Llinas. An introduction to multisensor fusion. In *Proceedings of the IEEE: Special Issues on Data Fusion*, pages 85(1):6–23, January 1997.

[34] A. Harma, M. F. McKinney, and J. Skowronek. Automatic surveillance of the acoustic activity in our living environment. In *IEEE International Conference on Multimedia and Expo*, Amsterdam, July 2005.

[35] J. Hershey, H. Attias, N. Jojic, and T. Krisjianson. Audio visual graphical models for speech processing. In *IEEE International Conference on*

Speech, Acoustics, and Signal Processing, pages V:649–652, Montreal, Canada, May 2004.

[36] J. Hershey and J. Movellan. Audio-vision: Using audio-visual synchrony to locate sounds. In *Advances in Neural Information Processing Systems*, pages 813–819. MIT Press, 2000.

[37] W. Hsu, L. Kennedy, C. W. Huang, S. F. Chang, and C. Y. Lin. News video story segmentation using fusion of multi-level multi-modal features in TRECVID 2003. In *International Conference on Acoustics Speech and Signal Processing*, Montreal, Canada, May 2004.

[38] K. S. Huang and M. M. Trivedi. Robust real-time detection, tracking, and pose estimation of faces in video streams. In *International Conference on Pattern Recognition*, Cambridge, United Kingdom, August 2004.

[39] V. Isler and R. Bajcsy. The sensor selection problem for bounded uncertainty sensing models. In *International Symposium on Information Processing in Sensor Networks*, pages 151–158, Los Angeles, CA, USA, April 2005.

[40] G. Iyengar, H. J. Nock, and C. Neti. Audio-visual synchrony for detection of monologue in video archives. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2003.

[41] G. Iyengar, H. J. Nock, and C. Neti. Discriminative model fusion for semantic concept detection and annotation in video. In *ACM International Conference on Multimedia*, 2003.

[42] R. Jain. Keynote speech on observation systems. In *The ACM International Workshop on Video Surveillance and Sensor Networks*, Singapore, November 2005.

[43] R. S. Jasinschi, N. Dimitrova, T. McGee, L. Agnihotri, J. Zimmerman, D. Li, and J. Louie. A probabilistic layered framework for integrating multimedia content and context information. In *International Conference on Acoustics, Speech and Signal Processing*, volume II, pages 2057–2060, Orlando, Florida, May 2002.

[44] O. Javed, Z. Rasheed, O. Alatas, and M. Shah. M-KNIGHT: A real time surveillance system for multiple overlapping and non-overlapping cameras. In *IEEE International Conference on Multimedia and Expo*, pages I:649–652, Baltimore, MD, USA, July 2003.

[45] F. V. Jensen. *Bayesian Networks and Decision Graphs*. Springer-Verlag, New York, USA, 2001.

[46] S. Jiang, R. Kumar, and H. E. Garcia. Optimal sensor selection for discrete event systems with partial observation. *IEEE Transactions on Automatic Control*, 48:369–381, March 2003.

[47] P. KaewTraKulPong and R. Bowden. An improved adaptive background mixture model for real-time tracking with shadow detection. In *European Workshop on Advanced Video Based Surveillance Systems*, London, UK, September 2001.

[48] M. Kam, Q. Zhu, and W. S. Gray. Optimal data fusion of correlated local decisions in multiple sensor detection systems. *IEEE Transactions on Aerospace and Electronic Systems*, 28(3):916–920, July 1992.

[49] L. A. Klein. *Sensor and Data Fusion Concepts and Applications*. SPIE Optical Engineering Press, second edition, 1999.

[50] M. G. Lagoudakis. The 0-1 KNAPSACK problem - An introductory survey. URL: citeseer.ist.psu.edu/151553.html.

[51] K.-Y. Lam, R. Cheng, B. Y. Liang, and J. Chau. Sensor node selection for execution of continuous probabilistic queries in wireless sensor networks. In *ACM International Workshop on Video Surveillance and Sensor Networks*, pages 63–71, NY, USA, October 2004.

[52] D. Li, N. Dimitrova, M. Li, and I. K. Sethi. Multimedia content processing through cross-modal association. In *ACM International Conference on Multimedia*, 2003.

[53] L. I.-K. Lin. A concordance correlation coefficient to evaluate reproducibility. *Journal of Biometrics*, 45(1):255–268, 1989.

[54] R. C. Luo, C.-C. Yih, and K. L. Su. Multisensor fusion and integration: Approaches, applications, and future research directions. *IEEE Sensors Journal*, 2(2):107–119, 2002.

[55] N. C. Maddage. *Content based music structure analysis*. PhD thesis, School of Computing, National University of Singapore, 2006.

[56] M. McHugh and A. F. Smeaton. Towards event detection in an audio-based sensor network. In *The ACM International Workshop on Video Surveillance and Sensor Networks*, pages 87–94, Singapore, November 2005.

[57] G. F. Meyer, J. B. Mulligan, and S. M. Wuerger. Continuous audio-visual digit recognition using N-best decision fusion. *Journal on Information Fusion*, 5:91–101, June 2004.

[58] A. V. Nefian, L. Liang, X. Pi, X. Liu, and K. Murphye. Dynamic bayesian networks for audio-visual speech recognition. *EURASIP Journal on Applied Signal Processing*, 11:1–15, 2002.

[59] C. Neti, B. Maison, A. Senior, G. Iyengar, P. Cuetos, S. Basu, and A. Verma. Joint processing of audio and visual information for multimedia indexing and human-computer interaction. In *International Conference RIAO*, Paris, April 2000.

[60] R. Nevatia, T. Zhao, and S. Hongeng. Hierarchical language-based representation of events in video streams. In *IEEE International Workshop on Event Mining*, Madison, Wisconsin, USA, June 2003.

[61] W. Niu, L. Jiao, D. Han, and Y. F. Wang. Human activity detection and recognition for video surveillance. In *IEEE conference on Multimedia and Expo*, Taiwan, June 2004.

[62] H. J. Nock, G. Iyengar, and C. Neti. Assessing face and speech consistency for monologue detection in video. In *ACM International Conference on Multimedia*, 2002.

[63] J. O'Brien. Correlated probability fusion for multiple class discrimination. In *Proceedings of Information Decision and Control*, pages 571–577, Adelaide, Australia, February 1999.

[64] J. B. Oommen and L. Rueda. A formal analysis of why heuristic functions work. *The Artificial Intelligence Journal*, 164:1–22, 2005.

[65] Y. Oshman. Optimal sensor selection strategy for discrete-time state estimators. *IEEE Transactions on Aerospace and Electronic Systems*, 30:307–314, April 1994.

[66] P. Pahalawatta, T. N. Pappas, and A. K. Katsaggelos. Optimal sensor selection for video-based target tracking in a wireless sensor network. In *IEEE International Conference on Image Processing*, pages V:3073–3076, Singapore, October 2004.

[67] I. Pavlidis and T. Faltesek. A video-based surveillance solution for protecting the air-intakes of buildings from bio-chem attacks. In *IEEE International Conference on Image Processing*, Rochester, New York, USA, September 2002.

[68] I. Pavlidis, P. Symosek, B. Fritz, M. Bazakos, and N. Papanikolopoulos. Automatic detection of vehicle occupants: The imaging problem and its solution. *Jounral of Machine Vision and Applications*, 11:313–320, 2000.

[69] J. O. Peralta and M. T. C. Peralta. Security PIDS with physical sensors, real-time pattern recognition, and continuous patrol. *IEEE Transanctions on Systems, Man, and Cybernetics - Part C: Applications and reviews*, 32(4):340–346, November 2002.

[70] D. G. Perez, G. Lathoud, I. McCowan, J.-M. Odobez, and D. Moore. Audio-visual speaker tracking with importance particle filter. In *IEEE International Conference on Image Processing*, 2003.

[71] A. Prati, R. Vezzani, L. Benini, E. Farella, and P. Zappi. An integrated multi-modal sensor network for video surveillance. In *The ACM International Workshop on Video Surveillance and Sensor Networks*, pages 95–102, Singapore, November 2005.

[72] B. S. Rao and H. D. Whyte. A decentralized bayesian algorithm for identification of tracked objects. *IEEE Transactions on Systems, Man and Cybernetics*, 23:1683–1698, 1993.

[73] M. Siegel and H. Wu. Confidence fusion. In *IEEE International Workshop on Robot Sensing*, pages 96–99, 2004.

[74] V. K. Singh, P. K. Atrey, and M. S. Kankanhalli. Coopetitive multi-camera surveillance using model predictive control. Technical report, School of Computing, National University of Singapore, April 2006.

[75] C. G. M. Snoek and M. Worring. Multimedia event-based video indexing using time intervals. *IEEE Transanctions on Multimedia*, 7(4):638–647, August 2005.

[76] M. Spengler and B. Schiele. Automatic detection and tracking of abandoned objects. In *Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, Nice, France, October 2003.

[77] H. Sridharan, H. Sundaram, and T. Rikakis. Computational models for experiences in the arts and multimedia. In *The ACM Workshop on Experiential Telepresence*, Berkeley, CA, USA, November 2003.

[78] C. Stauffer. Automated audio-visual activity analysis. Technical report, MIT-CSAIL-TR-2005-057, Massachusetts Institute of Technology, Cambridge, MA, USA, September 2005.

[79] C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 252–258, Ft. Collins, CO, USA, 1999.

[80] N. Tatbul, M. Buller, R. Hoyt, S. Mullen, and S. Zdonik. Confidence-based data management for personal area sensor networks. In *The Workshop on Data Management for Sensor Networks*, August 2004.

[81] A. Tavakoli, J. Zhang, and S. H. Son. Group-based event detection in undersea sensor networks. In *Second International Workshop on Networked Sensing Systems*, San Diego, California, USA, June 2005.

[82] V. Y. Teriyan and S. Puuronen. Multilevel context representation using semantic metanetwork. In *International and Interdisciplinary Conference on Modeling and Using Context*, pages 21–32, Rio de Janeiro, Brazil, February 1997.

[83] M. Valera and S. A. Velastin. Intelligent distributed surveillance systems: A review. *IEE Proceedings on Visual Image Signal Processing*, 152(2):192–204, April 2005.

[84] J. Wang, R. Achanta, M. S. Kankanhalli, and P. Mulhem. A hierarchical framework for face tracking using state vector fusion for compressed video. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Hong Kong, April 2003.

[85] J. Wang and M. S. Kankanhalli. Experience-based sampling technique for multimedia analysis. In *ACM International Conference on Multimedia*, pages 319 – 322, Berkley, CA, USA, November 2003.

[86] J. Wang, M. S. Kankanhalli, W.-Q. Yan, and R. Jain. Experiential sampling for video surveillance. In *ACM Workshop on Video Surveillance*, Berkley, November 2003.

[87] B. Wu, H. Ai, C. Huang, and S. Lao. Rotation invariant neural network-based face detection. In *IEEE Conference on Automatic Face and Gesture Recognition*, pages 79– 84, Seoul, Korea, May 2004.

[88] H. Wu. *Sensor Data Fusion for Context-Aware Computing Using Dempster-Shafer Theory*. PhD thesis, The Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA, December 2003.

[89] Y. Wu, E. Y. Chang, K. C.-C. Chang, and J. R. Smith. Optimal multimodal fusion for multimedia data analysis. In *ACM International Conference on Multimedia*, pages 572–579, New York, USA, October 2004.

[90] H. Xu and T.-S. Chua. Fusion of av features and external information sources for event detection in team sports video. *ACM Transactions on Multimedia Computing, Communications and Applications*, 2(1):44–67, February 2006.

[91] D. B. Yang and H. H. Gonzalez-Banos. Counting people in crowds with a real-time network of simple image sensors. In *IEEE International Conference on Computer Vision*, Nice, France, October 2003.