GOAL BASED OPTIMAL SELECTION OF MEDIA STREAMS

Pradeep K. Atrey and Mohan S. Kankanhalli

School of Computing, National University of Singapore
Republic of Singapore 117543
Email: {pradeepk, mohan}@comp.nus.edu.sg

ABSTRACT

A multimedia system utilizes a set of correlated media streams each of which partially help in achieving the system goal. However, since not all of the streams always contribute towards the goal, there is a need for determining the most informative subset from the available set of media streams at any instant. For example, a subset of two video cameras and two microphones could be better than any other subset of multimedia sensors at some time instance. This paper presents a novel framework to find the optimal subset of media streams that achieves the system goal under specified constraints. The proposed framework uses a dynamic programming approach to find the optimal subset of media streams based on two criteria; first, by maximizing the probability of achieving the goal under the specified maximum cost, and second by minimizing the cost of using the streams so that the goal is achieved with a specified minimum probability. To show the utility of our framework, we provide the simulation results for hypothesis testing.

1. INTRODUCTION

In recent times, it is being increasingly accepted that most media analysis tasks can be better performed by using multiple correlated media as compared to using only mono media. Examples include surveillance systems, media search systems and media mining systems. In surveillance systems, people employ multiple media such as video cameras, microphones and infra-red cameras, and also use other non-sensory data to achieve the system goals. The goal of a surveillance system could be ‘to monitor how many people have passed through the corridor between 4 pm to 5 pm on March 30, 2005’ or it could even be ‘to display the face of a person who shouted near the room number 101 in the corridor’. Using multiple media is advantageous because a single stream can only partially help in achieving a system goal due to its ability to sense only a part of the environment. Hence multiple media are used to capture different aspects of the environment to provide complementary information which is not available from only one single medium.

On the other hand, the fact that all of the employed streams do not always contribute towards a goal brings up the issue of finding best (or most informative) subset of streams. The system on the fly should be able to determine whether a particular subset of streams would be better than any other subset of streams. The best subset of media dynamically changes over time. Once this subset is found, the system can continue using it while ignoring the remaining streams for a certain period. This eliminates the cost of using redundant and less-informative media. The cost of using a media stream usually includes processing cost of stream, energy to operate the media device, and its wear and tear. Therefore, media must be used optimally so that a system goal can be maximally achieved under a specified cost.

In this paper, we essentially address the following research issues: 1) What is the optimal number of streams required to achieve the goal under the specified constraints? 2) Which subset of the streams is the optimal one? 3) Can one use alternate media streams without much loss of cost-effectiveness in case the most suitable subset is unavailable? 4) How frequently should this optimal subset be computed so that the cost of computing it can be minimized?

We study the problem of optimal media selection in two different ways - finding the optimal subset that maximizes the goal under a specified cost, and finding the optimal subset that minimizes the cost subject to goal being achieved to a certain given extent. We reduce the problem of optimal media selection to the 0-1 Knapsack problem [1] and use a dynamic programming approach to solve it. In our problem, for each media, the probability of it helping achieve the goal and its cost are analogous to profit and weight, respectively, of a knapsack problem. The fundamental difference is that we fuse the probabilities using a Bayesian approach [2], while the profits are added in 0-1 Knapsack problem.

In the past, optimal sensor selection problem has been widely studied in the context of discrete-event systems and failure diagnosis. The proposed approaches include optimal measurement subsystem strategy [3], Markovian decision strategy [4] and a formal method [5]. [3] and [5] do not consider the cost of using sensors. [4] assumes the uniform cost for all sensors which is impractical in a multimedia environ-
ment where different types of media are employed. A recently proposed method [6] selects the set of sensors based on a certain accuracy requirement. Our proposed work is different from all the above discussed solutions in that, our framework provide a tradeoff between the extent to which the system goal is achieved and the cost of using streams.

2. PROBLEM FORMULATION

Let a multimedia system $S$ designed for a goal $G$ employ a set $M^n(t) = \{M_1, M_2, \ldots, M_n\}$ of $n$ media streams at time $t$. For $1 \leq i \leq n$, $0 \leq p_i \leq 1$ be the probability of achieving the system goal $G$ using individual $i^{th}$ media stream. $P_i$ is also denoted as $P(G|M_i)$. Also, let $P_{\Phi}$ (also denoted as $P(G|\Phi)$) be the ‘fusion probability’ of achieving the system goal $G$ using a subset $\Phi \in $ (The power set of $M^n$) of media streams. The ‘fusion probability’ is the overall probability of achieving the system goal using a group of media streams [2]. For $1 \leq i \leq n$, $c_i$ be the cost per unit time of using the stream $i$. Also, $C_n = \sum_{i=1}^{n} c_i$ be the total cost.

We assume that - 1) All media capture the same environment (but optionally the different aspects) and provide correlated observations, 2) The system goal $G$ is to test a hypothesis $H$. The hypotheses could be ‘there is a person shouting near the meeting room’ or ‘there is currently a running person in a corridor’ etc, and 3) The number of streams is more than necessary to achieve the goal, hence there is a need to select the best subset.

The objective is to find a subset $\Phi \in \mathcal{P}(M^n)$ that -

**Problem MaxGoal**: maximizes $P_{\Phi}$ subject to $C_{\Phi} \leq C_{spec}$

**Problem MinCost**: minimizes $C_{\Phi}$ subject to $P_{\Phi} \geq P_{\Phi}^\text{spec}$

where $P_{\Phi}$ is the fusion probability of achieving the goal when the subset $\Phi$ of media streams is used by system $S$, $C_{\Phi}$ is the overall cost of using the subset $\Phi$ of streams, $P_{\Phi}^\text{spec}$ is the specified minimum fusion probability of achieving the goal, and $C_{spec}$ is the specified maximum overall cost.

3. PROPOSED FRAMEWORK

Given the set of $n$ media streams, the optimal subset of media streams to test a hypothesis $H$ is obtained as follows:

1. For $1 \leq i \leq n$, we first compute the probability $P(H|M_i)$ of hypothesis $H$ being true using a Bayesian classifier [7].

2. Using a voting strategy, we divide the $n$ streams into two subsets $S_1$ and $S_2$ based on whether at the current instant they support or do not support the true hypothesis.

3. For the two subsets $S_1$ and $S_2$, compute fusion probabilities $P(H|S_1)$ and $P(H|S_2)$ of achieving the goal using a Bayesian approach [2].

4. If $P(H|S_1) \geq P(H|S_2)$, we conclude that the hypothesis $H$ is true and find the optimal subset from $S_1$ using a dynamic programming approach, else the hypothesis is treated as null and the optimal subset is found from $S_2$.

3.1. Solution for MaxGoal

The dynamic programming approach for solving MaxGoal works as follows. We begin by considering the selection of $n^{th}$ stream. If we select the $n^{th}$ stream, then the fusion probability would be the result obtained from the fusion of $n^{th}$ stream with the remaining $n-1$ streams (with a maximum cost $C_{spec} - c_n$, where $c_n < C_{spec}$). However, if we do not select it, the fusion probability would possibly be the result obtained from the fusion of the remaining $n-1$ streams (with a maximum cost $C_{spec}$). The optimal fusion probability (of achieving the goal) will be the maximum of these two possible ‘best’ options. We describe the structure of an optimal solution by the following recurrence relation:

$$Prob(i,m) = \begin{cases} Prob(i-1,m) & , \ c_i > m \\ \max\{Prob(i-1,m), P_{\text{Fusion}}(Prob(i-1, m-c_i), p_i, \gamma_i)\} & , \ c_i \leq m \end{cases}$$

where $Prob(i,m)$, $1 \leq i \leq n$, $1 \leq m \leq C_{spec}$, is the optimal fusion probability (of achieving the goal) based on streams $1, 2, \ldots, i$ with the cost $m$. The initial conditions for the recursive relation are: \begin{align} Prob(1,m) = \begin{cases} 0 & , \ c_1 > m \\ p_1 & , \ c_1 \leq m \end{cases} \end{align}

The $P_{\text{Fusion}}$ function combines the probabilities of achieving the goal based on two sources $M^{-1}$ (i.e. a group of $i-1$ streams) and $M_i$ (i.e. an individual $i^{th}$ stream) using the following fusion model (described in [2]):

$$P_i = \frac{P_{i-1,p_i,e^{\gamma_1}}}{P_{i-1,p_i,e^{\gamma_1}} + (1 - P_{i-1})(1 - p_i),e^{-\gamma_i}}$$

where $P_i = \text{Prob}(i,m)$ and $P_{i-1} = \text{Prob}(i-1,m)$ are the probabilities of achieving the goal using $M^i$ and $M^{i-1}$, respectively. $p_i$ is the probability of $i^{th}$ stream individually helping achieve the goal, and $\gamma_i \in [-1,1]$ is the agreement coefficient between two sources $M^{i-1}$ and $M_i$. The limits -1 and 1 represent full disagreement and full agreement, respectively, between the two sources. The agreement coefficient between two sources is computed based on the cumulative past history of their agreement/disagreement. The detailed description of which is out of scope of this paper.

The algorithm MaxGoal outlines the idea described above.

**MaxGoal**($n$, $p$, $c$, $C_{\text{spec}}$, $\Gamma$)

**Inputs**

$n$: Number of input media streams.
$p_1[\ldots n]$: Probabilities of streams helping achieve the goal.
$c_1[\ldots n]$: Costs of using the streams.
$C_{\text{spec}}$: Specified maximum overall cost.
$\Gamma$: Set of agreement coefficients among the streams.

**Steps**

1. Initialize $Prob$ and $Select$ array to zero.
2. for $i = 1$ to $n$, $m = 0$ to $C_{\text{spec}}$
3. if $c[i] \leq m$
4. Compute fusion probability \( P_i \) using equation (1)
5. if \( P_i > \text{Prob}[i-1, m] \)
6. \( \text{Prob}[i, m] = P_i; \text{Select}[i, m] = 1 \)
7. else
8. \( \text{Prob}[i, m] = \text{Prob}[i-1, m]; \text{Select}[i, m] = 0 \)
9. else \( \text{Prob}[i, m] = \text{Prob}[i-1, m]; \text{Select}[i, m] = 0 \)
10. \( K = m - 1, P_S = \text{Prob}[n, K], C_\Phi = 0 \)
11. for \( i = n \) to 1 in steps -1
12. if \( \text{Select}[K] = 1 \)
13. Output the stream \( i \) into \( \Phi \)
14. \( C_\Phi = C_\Phi + c[i], K = K - c[i] \)

**Outputs:** \( \Phi, P_\Phi \) and \( C_\Phi \).

### 3.2. Solution for MinCost

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 optimal cost of using remaining \( n - 1 \) streams so that the overall probability of achieving the goal is at least \( \text{Prob}_\text{spec} \). However, if we don’t select it, then the best cost would possibly be the cost of using the remaining \( n - 1 \) streams. The optimal probability of achieving the goal will be the minimum of these two potentially ‘best’ options.

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

\[
\text{Cost}(i, m) = \begin{cases} 
\min(\text{Cost}(i - 1, m), c_i), & m \leq \min(p_i, \text{Prob}_\text{spec}) \\
\min(\text{Cost}(i, m), \text{Cost}(i - 1, m)), & m > \min(p_i, \text{Prob}_\text{spec}) \\
\min(\text{Cost}(i - 1, m), \text{Cost}(i, m)), & m = \min(p_i, \text{Prob}_\text{spec}) \\
\text{Cost}(i - 1, m), & m > \text{R}'
\end{cases}
\]

where \( 1 \leq i \leq n, 1 \leq m \leq L \). The initial conditions are,

\[
\text{Cost}(1, m) = \begin{cases} 
c_1, & m \leq \min(p_1, \text{Prob}_\text{spec}) \\
\infty, & m > p_1
\end{cases}
\]

In the recursive formulation described above, \( f_{cost} \), \( R \) and \( R' \) are computed as follows,

\[
f_{\text{cost}} = \begin{cases} 
\text{Cost}(i - 1, l[s]), & s > 0 \text{ and } l[s] \neq p_i \\
c_i, & s > 0 \text{ and } l[s] = p_i \\
0, & s = 0
\end{cases}
\]

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

\[
R' = \begin{cases} 
\max(R', R), & s > 0 \\
0, & s = 0
\end{cases}
\]

\( l[s] \) is a temporary array that contains the individual streams’ probabilities as well as their fusion probabilities.

We have also developed the corresponding MinCost algorithm, which is structurally similar to MaxGoal. Its description has been omitted due to space constraints.

### 3.3. Complexity analysis

Any brute-force approach to solve MaxGoal takes \( O(2^n) \) time since all the \( 2^n \) subsets of streams are checked to find the optimal subset. However, the time complexity of MaxGoal algorithm is \( O(n^2 \times \text{Prob}_\text{spec}) \) which is on average lower than the brute-force approach. Note that \( O(n^2 \times \text{Prob}_\text{spec}) \) also includes the time complexity of PFusion, which is \( O(n) \).

The space complexity of the MaxGoal is \( O(n \times \text{Prob}_\text{spec}) \).

The algorithm MinCost has a time complexity of \( O(n^2 \times L) \) to find the optimal subset which is again better than the brute-force approach. Note that higher the discrete levels \( L \) of probability value, higher would the time complexity be.

The space complexity is \( O(n \times L) \).

Note that these time and space complexities are of polynomial time under the condition that \( \text{Prob}_\text{spec} \neq 0 \) (for MaxGoal) and \( L \neq 0 \) (for MinCost).

### 4. SIMULATION RESULTS

We provide the simulation results for a set of 10 media streams with individual probabilities of hypothesis being true and the cost 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 \( c = (9, 4, 2, 8, 2, 8, 5, 2, 3) \), respectively. First, the streams are divided into two sets \( S_1 \) and \( S_2 \) based on whether or not they support the true hypothesis. Precisely, the streams that support the hypothesis with more than 0.50 probability 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 \( S_2 = (0.55, 0.60, 0.55, 0.70) \). Note that, after this division, the sets \( S_1 \) and \( S_2 \) support true hypothesis and null hypothesis, respectively. Next, we fuse the streams from two sets individually and obtain the fusion probabilities \( P(H|S_1) \) and \( P(H|S_2) \) (Refer to Table 1). For sake of simplicity, we have assumed uniform agreement coefficient among all the media streams. However, we analyze how the system behaves by having different values (0.00, 0.50 and 1.00) of this uniform agreement coefficient. As shown in Table 1, \( P(H|S_1) \) is higher than \( P(H|S_2) \); this suggests that the hypothesis is true. 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 \( \text{Prob}_\text{spec} \) and the specified minimum probability \( \text{Prob}_\text{spec} \) of achieving the goal, respectively. The simulation results of MaxGoal and MinCost are shown in figure 1a-1b and figure 1c-1d, respectively. In figure 1a-1d, symbols A, B, and so on, represent the optimal subsets. For instance, in figure 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 \( \Phi \) 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
For instance, figure 1a clearly shows that the subset indicated by symbol F achieves a lower probability when the agreement coefficient between the streams is low (the symbols C and D).

The overall observations from simulation (figure 1) are:

1. The proposed framework 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 1a clearly shows that the subset indicated by symbol E would be a better choice over the subset indicated by symbols H onwards 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 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 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. The absolute difference of fusion probabilities \( P(H|S_1) \) and \( P(H|S_2) \) of two sets \( S_1 \) and \( S_2 \) suggests how sure are we about the correct testing of hypothesis \( H \) (Table 1). If this difference is significant, it is reasonable to continue with the current optimal subset for a certain period. However, if it is low, then the optimal subset should be recomputed more frequently since it could be more likely that one or two streams may switch to the other set. This gives us some idea on how frequently the optimal subset should be computed. However this need to be formally proven.

4. 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.

### 5. CONCLUSIONS

In this paper, we propose a framework that uses a dynamic programming approach to find the optimal subset of media streams for two different objectives - maximizing the probability of achieving the goal under the specified cost, and minimizing the cost of using the subset to obtain a specified probability of achieving the goal. The simulation results show that the dynamic programming based approach provides the best subset under specified constraints and it also offers the user a flexibility to choose alternative (or next best) subset when the best subset is unavailable. For future work, it would be interesting to see how the accuracies of media streams can be incorporated into our framework.

### 6. REFERENCES


