MULTIMEDIA DECISION FUSION

XIANGYU WANG

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF SCHOOL OF COMPUTING

NATIONAL UNIVERSITY OF SINGAPORE

2012

Acknowledgments

This thesis is the result of five 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.

My most earnest acknowledgement must go to my supervisor, Prof. Mohan S. Kankanhalli, who has been instrumental in ensuring my academic, professional, financial, and moral well being ever since. He has supported me throughout my thesis with his patience and knowledge. I attribute the level of my Ph.D. degree to his encouragement and effort. This thesis would not have been completed or written without him. I could not have imagined having a better or friendlier supervisor for my Ph.D.. During the five years of my Ph.D., I have seen in him an excellent supervisor who can bring the best out from his students, an outstanding researcher who is insightful and can constructively criticize research, and a nice human being who is honest, fair, and helpful to others.

I sincerely thank Prof. Min-Yen Kan, Prof. Low, Bryan Kian Hsiang, and Prof. Ye Wang for serving on my doctoral committee. Their constructive feedback and comments at various stages have been significantly useful in shaping the thesis up to completion. I would also like to thank the other member of my oral defence committee, Prof. Jun Wang, for his time and insightful comments.

My sincere thanks go out to Dr. Yong Rui and Prof. Pradeep K. Atrey with whom I have collaborated during my Ph.D. research. Their conceptual and technical insights into my thesis work have been invaluable.

After accomplishing undergraduate study, 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.

In my daily work I have been blessed with a friendly and cheerful group of fellow

students. My time at NUS was made enjoyable in large part due to the many friends and groups that became a part of my life. I am grateful for time spent with friends.

I would like to thank my family for all their love and encouragement. My parents deserve special mention for their inseparable support and prayers. My Father, Shigong Wang, in the first place is the person who has built the foundation for my learning character by showing me the joy of intellectual pursuit ever since I was a child. My Mother, Qinglan Ma, is the one who sincerely raised me with her caring and gentle love. Words fail me to express my appreciation to my wife Lu Yang whose dedication, love and persistent confidence in me, has taken the load off my shoulder. I would also thank her family for letting me take her hand in marriage, and accepting me as a member of the family, warmly.

Finally, I would like to thank everybody who was important to the successful realization of thesis, as well as expressing my apology that I could not mention personally one by one. Thank you.

Contents

Summary

The amount of multimedia data available on the Internet has increased exponentially in the past few decades and is likely to keep on increasing. Given multimedia's nature of having multiple information sources, fusion methods are critical for its data analysis and understanding. Multimedia fusion is a way to integrate multiple media, their associated features, or the intermediate decisions in order to perform an analysis task. It is useful for several objectives such as detection, recognition, identification, tracking, and decision making in many application domains. Multimedia fusion has been attracting increasing attention. However, some important issues in the multimedia fusion still need to be properly studied, such as how to utilize the correlation among different multimedia information sources, how to cope with the uncertainty and diversification of multimedia information, and how to adapt the fusion models to the conditions of changing and increasing amount of data.

This thesis proposes fusion methods that address the research challenges of proper utilization of the correlation among multimedia information sources. The thesis also addresses how to evolve the multimedia fusion model and improve the performance with new data. In MultiFusion, we make more use of the correlation among multimedia information sources by combining and utilizing the correlation in each iteration of an Adaboost-like structure. In portfolio fusion method, we maximize the return and minimize the risk (uncertainty) to achieve a high dependable performance by introducing the widely used and effective portfolio theory from finance. A more sophisticated model to utilize correlations among different information sources is also presented. For the situation that the multimedia data keep increasing with time and the nature of the data collection can change, we develop the Up-Fusion method. With the utilization of multimedia correlation and refinement, the method evolves the fusion model along with the newly added multimedia data to improve the performance. Moreover, the situations

that the labels of newly added data are not available and that the context or nature of data changes, are also handled by using pseudo labels and sliding window. How to fuse the information sources most appropriately is also considered in this thesis. Based on the common practice of seeking opinions from specialists before making a decision, a specialist fusion method that adaptively predicts the expertise of different information sources on different data instances and effectively combines the expertise with decision is proposed in this thesis. The proposed fusion methods are mainly intended for classification and retrieval problems which are the main problems of multimedia applications.

To show the advantages and utility of our methods, simulation and real application experimental results are provided for each fusion method. Moreover, the fusion methods in the thesis aim to solve different objectives. The appropriate situations for different fusion methods are argued in the conclusion chapter. In the end, some limitations and broad vision for multimedia fusion methods are discussed.

List of Figures

List of Tables

Chapter 1

Introduction

With the advances of technology and ubiquitous spread of multimedia devices, the number of multimedia applications has been increasing over the past two decades. Consequently, the sources of multimedia data production are also proliferating at an unprecedented pace. The amount of multimedia data available on the Internet has increased exponentially, such as broadcast news archives, radio recordings, music collections, TV program archives, lecture and presentation recordings, meeting room recordings, and personal archives. With the enormous amount of multimedia data, it is inefficient and tedious to manually analyze the data. In order to facilitate the use of the multimedia data, analysis of multimedia data is therefore needed in many applications such as information retrieval, education, and security. A multimedia analysis task involves processing of multimodal data in order to obtain valuable insights about the data, a situation, or a higher level of activity [Atrey, Kankanhalli, and Jain, 2006]. For example, surveillance systems utilize the data from multiple types of sensors like microphones, video cameras to detect events such as bag abandonment; for news video retrieval, video data are combined with audio data and text information for concept detection to enable semantic search. Multimedia fusion is a way to integrate multiple media, their associated features, or the intermediate decisions in order to perform an analysis task.

Given multimedia's nature of having multiple information sources, fusion methods are critical for its data analysis and understanding. Multimedia fusion is useful for several objectives such as detection, recognition, identification, tracking, and decision making in many application domains.

1.1 Overview of multimedia fusion

1.1.1 Motivation of multimedia fusion

Multimedia data generally comprise of data of different modalities. Here, a media is characterized mostly by its nature (for example, audio, video, and text), while a modality is characterized by both its nature and the physical structure of the provided information (for example, X-Ray image, and MRI(Magnetic resonance imaging) image). In other words, the multimedia data usually contain several different information sources. Data of different modalities are obtained from different information sources, and the useful information can be extracted from the data with proper analysis. For example, surveillance data usually contain video data captured by regular camera and infrared camera as well as audio data; the photo data from Flickr contain image content as well as text tags and descriptions; the multimedia digital library data contain video, audio, and text. However, conventional analysis methods generally utilize one information source while leaving out the other information sources. Nevertheless, no single information source can help to accomplish the analysis task perfectly. As a result, people seek ways to combine the different information sources to improve the performance using the complementary, correlated or redundant data available in these information sources. Take a surveillance system for example: The "running" event can be detected by both visual and audio data. But neither contains complete information for the task. The event cannot be accurately decided with either moving speed from visual data or sound from audio data. The result can be improved by taking both information sources. In [Yang

et al., 2007], visual low-level features, semantic features, audio feature, and surrounding text features are fused for better web video categorization. In [Geng et al., 2010], gait and face information is fused for robust human identification. Thus, multimedia fusion is quite useful and has attracted much attention. In general, the multimedia fusion procedure can be illustrated as in Figure 1.1.

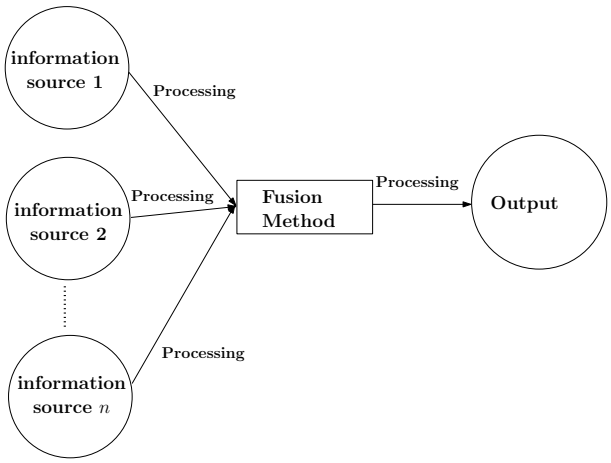


Figure 1.1: The illustration of the multimedia fusion framework

"*The intrinsic connection to our daily life experiences provides an undeniably strong psychological pretext. Seeking additional opinions before making a decision is an innate behavior for most of us, particularly if the decision has important financial, medical or social consequences. Our goal in considering the decisions of multiple experts, is to improve our confidence that we are making the right decision, by weighing various opinions, and combining them through some thought process to reach a final decision.* [Polikar, 2006]." For example in financial security investment, instead of thinking about one major factor or consulting one person, people would like to take all the related factors into account or seek different people's opinions in order to make a good decision.

There are also several mathematically sound reasons for considering multimedia fusion. A set of information sources may have different performances. Combining the outputs of several information sources may reduce the risk of an unfortunate selection of

a poorly performing information source. The fusion of information sources may or may not beat the performance of the best information source in the ensemble, but it definitely reduces the overall risk of making a particularly poor selection. In multimedia fusion, if we have several sets of data obtained from various information sources, it should be helpful to fuse different information sources instead of using one single information source.

1.1.2 Advantages of multimedia fusion

Essentially, multimedia fusion method is useful because the multimedia fusion utilizes more information sources, hence more information, than single modality data analysis.

- First, the different multimedia sources contain correlated information. The correlations can be complementary, redundant, or a mix. For example, there are several cameras in the surveillance systems. If two cameras are capturing different perspectives of the environment, the information of the two cameras will be complementary. If two cameras are capturing almost the same perspective, the information will be redundant (the redundancy here means the two cameras will give almost the same decisions for surveillance). If two cameras are capturing the environment at different but overlapped environment, the information will be in between of complementary or redundant. The information from any single multimedia source is usually incomplete. It improves the fusion results by making good use of the correlations among different information sources. Recall the surveillance system example. The "running" event can be detected by both visual and audio data. But neither contains complete information for the task. The event cannot be accurately decided with either moving speed from visual information source or sound from audio information source. People can walk fast or run quietly. The result can be improved by taking both information sources.

- Second, the fusion of several information sources may reduce the risk of an unfortunate selection of a poorly performing information source. For example, the regular camera works well in the day, while infrared camera works better in the dark. Fusion of the data from two cameras will reduce the risk of poor performance at any time.

1.1.3 Levels of multimedia fusion

Generally speaking, there are three different fusion categories: low, intermediate, or high level fusion, depending on the processing stage at which fusion takes place [Dasarathy, 1994].

- Low level fusion, also called *data level fusion*, combines several sources of raw data to produce new raw data that are expected to be more informative than the inputs. For example, regular image and infrared image are fused to enhance photos [Zhang, Sim, and Miao, 2008]. The data fusion combines and utilizes the raw data which contain the comprehensive information. However, the multimedia data are usually heterogeneous. Hence, it is difficult to integrate the multimedia data.

- Intermediate level fusion, also called *feature level fusion*, combines various features to produce a better feature set. Those features may come from several raw data sources (several sensors, different moments, *etc.*) or from the same raw data source. The objective is to obtain a limited number of relevant features. For example, Snoek *et al.* in [Snoek, Worring, and Smeulders, 2005] proposed the classification-based feature fusion method. The method concatenates unimodal feature vectors to obtain a fused multimedia representation and then relies on supervised learning to classify semantic concepts.

- High level fusion, also called *decision level fusion*, combines decisions coming from

several sources. By extension, one speaks of decision fusion even if the decision is a confidence score (soft decision) and not a crisp decision (non-fuzzy decision, e.g., "yes / no"). For example, Geng *et al.* [Geng et al., 2010] proposed a context-aware fusion method. Instead of using static fusion weights, the method uses linear weighted sum method and introduces context factors to dynamically adapt the weights to the environment.

Data fusion can, due to the data processing inequality, achieve the best performance improvements, because at this early stage of processing the most information is available [Koval, Voloshynovskiy, and Pun, 2007]. Complex relations in data can be exploited during fusion, provided that their way of dependence is known. In practice, the exploitation of feature or modality dependencies presumes their statistical knowledge, which can be problematic. Drawbacks in data and feature fusion are problems due to the "curse of dimensionality", its computational expense and that it needs a lot of training data. The opposite is true for decision fusion [Kludas, Bruno, and Marchand-Maillet, 2007]. In multimedia applications, the data are usually of different modalities. It is very difficult to combine raw data or features from different modalities. The decision fusion generally has the best portability. Moreover, according to the work of Snoek *et al.* [Snoek, Worring, and Smeulders, 2005], the classification-based decision fusion method tends to give better performance than the classification-based feature fusion method in multimedia applications.

1.1.4 Strategies of multimedia fusion

In the fusion of complementary information, the information gain results from combining multiple complementary information sources to generate a more complete representation of the world. Here, the overall goal is to exploit the sources' diversity or complementarity in the fusion process. In the fusion of redundant information, the fusion method utilizes

the redundancy in information sources. It provides a reduced overall uncertainty and hence also increased robustness in fusion systems by combining multiple information sources or, *e.g.*, multiple features of a single source [Kludas, Bruno, and Marchand-Maillet, 2007]. Given two vectors X, Y, a measure of the exchange of information, called *mutual information*, is denoted as $I(X, Y)$. The conditional entropy that permits to measure the additional information from the vector Y given the vector X is denoted as $H(Y|X)$. The information-theoretic description provides thus a representation of the dual concepts of redundancy and complementarity. In fact, we have $I(X, Y)+H(Y|X) = H(Y) =$ constant. It is shown as in Figure 1.2. Since the sum of complementarity and

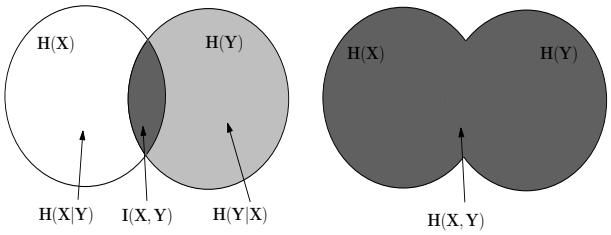


Figure 1.2: Venn diagram: the relationship between simple entropy, joint entropy and mutual information

redundancy of a source equals a constant, it is only possible to optimize a fusion system in favor of the one or the other [Fassinut-Mombot and Choquel, 2004].

The various fusion methods are generally divided into three categories: Rule-based methods, Classification-based methods, and Estimation-based methods [Atrey et al., 2010]. This categorization is based on the basic nature of these methods. Rule-based methods include Linear Weighted Fusion, Majority Voting Rule, and Custom-defined rule. Classification-based methods include Support Vector Machine, Bayesian Inference, Dynamic Bayesian Networks, Neural Networks, and Maximum Entropy Model. Estimation-based methods include Kalman Filter, Extended Kalman Filter, and Particle Filter.

1.1.5 Issues of multimedia fusion

Several factors need to be considered when designing multimedia fusion methods:

- **Fusion quality**: Fusion of several multimedia information sources may achieve the desired output or a worse output. Some issues such as, how to utilize the correlations among multimedia information sources, how to properly fuse the information from different multimedia information sources, *etc.*, should be considered to obtain the desired fusion output.

- **Scalability**: The multimedia fusion method may work well for certain number of multimedia information sources. However, whether it can still work or can be easily adapted when the number of multimedia information sources increases is a desired property.

- **Portability**: Multimedia fusion methods can be ad-hoc or generic. It is better to generalize the fusion method so that the method can be applied into different application scenarios. In multimedia applications, the data are usually heterogeneous. Thus, the portability of data and feature level fusion is not easy to achieve, while decision level fusion can usually have good portability since different information sources can be analyzed using different yet appropriate methods to obtain the individual decisions with the same representation.

- **Computational Complexity**: Different applications may have different requirements on the computation complexity of multimedia fusion method. Real-time application may require a fast computational fusion method, while off-line processing may not care much about that. It is always desirable to have a light computation multimedia fusion method, but it usually is a compromise between computational complexity and fusion quality.

1.2 Advantages of Multimedia decision fusion

Decision level fusion is suitable for multimedia data analysis because of the following reasons:

- In decision level fusion, data from different information sources can be analyzed using different yet appropriate methods to obtain the individual information source decisions having the same representation. For example, in multimedia sensor surveillance, video and audio from sensors can be processed separately to extract different features at different sampling rates. These features can be then processed using different methods, such as learning algorithm or rules, to get the decisions. It is easy to combine the homogeneous decisions, while it is hard to combine the heterogeneous data/feature. Data/feature level fusion needs perfect synchrony and cannot perform well when the nature of data/feature is different. Decision level fusion provides much more flexibility to the multimedia fusion process.

- Decision level fusion is intuitive and easy to perform, while data/feature level fusion suffers from "*curse of dimensionality*". The work of Beyer *et al.* [Beyer et al., 1999] shows that, when data dimensionality is large, the distances between pairs of objects in the space become increasingly similar to each other due to the *central limit theorem*. This phenomenon is called the *dimensionality curse* [Bellman, 1961], because it can severely hamper the effectiveness of data analysis [Wu et al., 2004]. In [Yanagawa et al., 2007], many high-dimensional features, such as edge direction histogram (73D), Gabor textures(48D), and grid color moments(225D), are used for concept detection. The dimensionality will be very large if feature level fusion is employed.

- In decision level fusion, it is easy to control the relative contributions of information sources to fusion results (*e.g.*, by weighting), while in data/feature level

fusion, this is more difficult.

- Though the information contained in a decision is less, we can use confidence score to keep the possible hypotheses in decision level fusion. It is consistent with *Principle of Least Commitment* (Don't do something that may later have to be undone. To keep multiple hypotheses alive for subsequent processing until a crisp decision is required [Keller, Gader, and Caldwell, 1995]). Moreover, according to *Principle of Graceful Degradation* (degrading the data should not prevent the delivery of at least some of the answer), useful information can still be retained in decisions.

- When a new information source is introduced, decision level fusion only needs to train the model for the new information source as well as fusion model. Thus, it is easy to scale. But the whole model needs to be trained again for data/feature level fusion.

The comparison of data, feature and decision level fusion is summarized in Table 1.1.

Category	**Dimensionality**	**Data of different nature**	**Information**	**Scalability**
data level fusion	high	hard to combine and cannot perform well	the most information	difficult
feature level fusion	high	hard to combine and cannot perform well	less	difficult
decision level fusion	low	can use different yet appropriate methods	least	easy

Table 1.1: Comparison of different fusion levels

1.3 Scope of the dissertation

There are some works on related topics such as *Multi-Classifier Systems* (MCSs) (also known as ensembles or committee of classifiers). Slightly different classifiers can be obtained by using different learning paradigm for the approximation of the same function.

Such a combination of diverse classifiers for the combined classification of data is known as a multi-classifier.

Multimedia fusion is similar to MCS in that they both fuse information to improve the results. However, compared to Multi-Classifier Systems, multimedia fusion has three main differences:

- First, multimedia fusion is to combine multiple information sources, *e.g.*, multiple modalities or by extension, multiple features of a single modality. The data are usually heterogeneous. The Multi-Classifier Systems will just utilize different learning methods on the same data. Diversity for Multi-Classifier Systems results from independence among classifiers. The diversity of multimedia fusion is more complex. It also results from different information sources.

- Second, multimedia fusion can adopt fusion at different levels: data level, feature level, or decision level. The Multi-Classifier Systems have just decision fusion of different classifiers. The decisions can be obtained using classification paradigms while it is not necessary for multimedia fusion.

- Third, multimedia fusion may be affected more by context compared to Multi-Classifier Systems. Multimedia fusion combines multiple heterogeneous information sources on which context may have different effects. Multi-Classifier Systems use the same data and may not be greatly affected by the context.

These differences make multimedia fusion, especially multimedia decision fusion, an equally, if not more, challenging problem. In this dissertation, we will focus on the multimedia decision fusion problem, especially the decision fusion strategies and their corresponding multimedia applications, mainly classification and retrieval. The Multi-Classifier Systems are out of the scope of this dissertation, but some of the ideas may be useful for multimedia decision fusion and will be mentioned.

1.4 Summary

Multimedia fusion presents new opportunities because more complete and diverse information can be extracted through facets of multimedia sources such as speech transcript text, audio, camera motion, and visual features. The multimedia fusion approach is becoming more useful as multimedia data proliferates. The multimedia fusion problem can be represented in a general formulation as follows:

$$I = \mathcal{F}(I_1, \ldots, I_n) \tag{1.1}$$

where I_i is the decision from information source i, I is the final fusion decision, and $\mathcal{F}$ is the fusion function. Several decision fusion methods have been proposed in literature. However, some important issues in the multimedia fusion still need to be properly studied. The objective of this dissertation is to develop fusion methods that address various research challenges. In this thesis, the literature of multimedia decision fusion methods have been carefully studied and reviewed. Several multimedia decision fusion methods have been proposed to solve some important issues encountered in previous studies:

- First and foremost, the correlation among different information sources is not well utilized to obtain better results. The different information sources contain complementary, redundant, or correlated information. There are different goals in fusion of different correlation information. In the fusion of complementary information, the information gain results from combining multiple complementary information sources to generate a more complete representation of the world. Here, the overall goal is to exploit the sources diversity or complementarity in the fusion process. In the fusion of redundant information, the fusion method utilizes the redundancy in information sources. It provides a reduced overall uncertainty and hence also increased robustness in fusion systems by combining multiple information sources

or, *e.g.*, multiple features. However, how to differentiate the correlation among information sources and how to utilize the different correlations are seldom considered. A well utilization of correlation among information sources should improve the fusion performance significantly. Most of the previous methods generally combines different information sources only once, like [Polikar et al., 2001]. We try to apply fusion and use correlation multiple times in MultiFusion method proposed in Chapter 3. By using boosting structure and combining the results at each iteration, the fusion performance is improved.

- The uncertainty in decision is occasionally considered. The decision cannot be estimated with absolute certainty using the classification models. The uncertainty is the lack of complete certainty, that is, the existence of more than one possibility. There are many sources of uncertainty such as ambiguity, noise, and deviations between the scoring function and the true probability of relevance. Thus, the risk (uncertainty) is an intrinsic feature of prediction using classification models. Taking the real accuracy as type of "an investment return" of our classification models, we should maximize the return as a desirable thing and minimize the variance of the return as an undesirable thing. Most of the methods consider the fusion as an information aggregation task. They aim to maximize the aggregated information by assigning proper weights to individual information channels [Li et al., 2009], for example, Max, Min, Average [Ngo et al., 2007] fusion methods. To the best of our knowledge, minimizing the effect of uncertainty has never been explicitly considered in multimedia fusion methods. Thus, how to minimize the uncertainty while maximizing the return should be studied. Sophisticated correlation model should also be helpful in this. We try to propose a decent formalized method to consider the uncertainty and find the optimized weights. A portfolio fusion method is proposed in Chapter 4 by introducing the widely used

and effective portfolio theory from finance.

- In multimedia fusion, the evolution of the fusion model is of primary importance because of the nature of multimedia applications. First of all, the semantic label information is important for multimedia analysis because many multimedia analysis tasks are based on classification and a large amount of labeled training data are necessary for good classification. However, most of multimedia data have limited label information, or even worse, has no label information. For example, on Flickr, the label for the multimedia document (image, tags and description) is not available or quite noisy. Labeled examples are fairly expensive to obtain due to the high labor costs faced when annotating videos [Wang et al., 2007]. Thus, little amount of training data are available at the beginning. The fusion performance may suffer as a result. Furthermore, the multimedia data keep increasing with time. New instances of multimedia data are continuously added. For example, new videos are periodically uploaded to Youtube. Thus, the fusion model may not always be valid or effective as the multimedia data increase because the nature of the data collection can change. As a result, it will be quite useful to evolve the multimedia fusion model and improve the performance with new data. The previous methods generally cannot cope with the new data well. For example, the context aware fusion methods like [Movellan and Mineiro, 1998], [Lee and Park, 2008], [Geng et al., 2010] need the context information which may not be available and dealing with all influential context factors is unrealistic in practice. An evolving fusion method, called Up-Fusion, is proposed in Chapter 5.

- The confidence measurement of individual output decision should be considered. The output from each individual information source may not reflect the confidence. Moreover, the information source may have different confidence on different output so that an overall weight for the information source is not suitable. Some methods

like [Keller, Paterson, and Berrer, 2000; Brazdil and Soares, 2000] have adopted data dependent combination. However, the expertise of individual information source has not been exploited in multimedia fusion. Thus, how to measure the expertise of the output from individual information source and then effectively adopt it in fusion process is also an important problem. A specialist fusion method is proposed in Chapter 6 by measuring expertise of different information sources on different data.

The fusion function $\mathcal{F}$ varies with different fusion methods. For MultiFusion, $\mathcal{F}$ is a weighted majority voting function. For portfolio fusion, $\mathcal{F}$ is a linear function. For upfusion method, $\mathcal{F} = \mathcal{F}_t$ is evolved when new data are added with time and for each iteration it is a linear function. For specialist fusion, $\mathcal{F} = \mathcal{F}_{\mathbf{X}}$ is a data dependent linear function.

The dissertation is organized as follows: Chapter 2 categorized and reviewed the literature of multimedia decision fusion methods according to the nature of combination strategies. Chapter 3 proposed a method to apply fusion and explicitly use correlation multiple times. By using boosting structure and combining the results at each iteration, the fusion performance is improved. Chapter 4 proposed a decent formalized method to consider the uncertainty and find the optimized weights for linear fusion. Chapter 5 proposed a method to cope with the situation that the multimedia data keep increasing. The method evolves the multimedia fusion model and improves the performance with new data. Chapter 6 adaptively combined the decisions from different information sources by measuring expertise of different information sources on different data.

Chapter 2

Literature Review

There have been several studies on the multimedia fusion problem in the literature. Various ways of combining the evidences from different information sources have been proposed. Information fusion is defined as "an information process that associates, correlates and combines data and information from single or multiple sensors or sources to achieve refined estimates of parameters, characteristics, events and behaviors" [Llinas et al., 2004]. The roots of decision fusion can be found in the neural network literature, where the idea of combining neural network outputs was published as early as 1965 [Nilsson, 1965]. Later its application expanded into other fields like econometrics as forecast combining, machine learning as evidence combination and also information retrieval in *e.g.* page rank aggregation [Kludas, Bruno, and Marchand-Maillet, 2007].

There is a good survey on multimodal fusion for multimedia analysis [Atrey et al., 2010]. It provides an overview of fusion strategies of different levels. This chapter adopts some of the categorization and focuses on the multimedia decision fusion. The state-of-the-art literature that uses different multimedia decision fusion strategies for various analysis tasks such as audio-visual person tracking, video summarization, multimodal dialog understanding, speech recognition and so forth is commented. Various issues such as the use of correlation, context and confidence, and the optimal modality selection that

influence the performance of a multimodal fusion process are also critically discussed.

The classification systems are different according to different facets of fusion methods. According to the variability of the fusion strategies, the fusion methods can be divided into static and non-static categories. In the static fusion, the fusion rules are predefined and remain fixed when the system is running [Geng et al., 2010]. On the contrary, the fusion rules in non-static fusion can evolve as the system running. According to the process of algorithms, fusion can be divided into two types, non-heuristic and heuristic [Tan et al., 2009]. Non-heuristic algorithms do not need the training phrase. Through a simple calculation, such as Max, Min, Average, and Product, *etc.*, they can get the results. Non-heuristic algorithms are simple but not efficient. Heuristic algorithms include some parameters, and require special data sets for training. Such algorithms are OWA (Ordered Weighted Average) [Yager, 1988], WA (Weighted Average) [Wu and Crestani, 2002], and so on. According to the nature of combination strategies in the fusion methods, the decision fusion methods can be categorized into three categories: Rule-based methods, Classification-based methods, and Cascaded (or Sequential) methods. Similar to [Atrey et al., 2010], rule-based methods denote the methods that combine information sources using different rules, which include Linear Opinion Pool, Independent Opinion Pool, Bayesian rule, min-max rule, majority voting rule, and Dempster-Shafer (D-S) Theory. Classification-based methods represent the methods that combine different information sources and obtain the decision based on classification models. The methods in this category include classification-based decision fusion like super-kernel fusion [Wu et al., 2004]. Cascaded methods are the methods combine multimedia information sources sequentially instead of fusing together at the same time. This kind of methods usually uses filtering or boosting methods. In the remainder of this chapter, the nature of combination strategies categorization will be mainly adopted and the fusion methods are discussed separately. The rule-based fusion methods are discussed in Section 2.1, the classification-based fusion methods are

discussed in Section 2.2, and the cascaded fusion methods are discussed in Section 2.3. Finally, the discussion is given in Section 2.4.

2.1 Rule-based Fusion

2.1.1 Overview of methods

Various rule-based fusion methods have been proposed to use rules to combine multimedia data, such as Linear fusion, Dempster-Shafer theory, maximum rule, minimum rule, median rule, and (weighted) majority vote.

2.1.1.1 Linear Opinion Pool

When a group of N individuals are required to make a joint decision, it occasionally happens that there is an agreement on a utility function for the problem but that opinions differ on the probabilities of the relevant states of nature. Stone in [Stone, 1961] proposed a fusion rule by attaching a measure of value such as weight to the information provided by each information source.

Suppose that,

- Y is the set of available decisions
- X denote the state of nature, to which probability density functions on some measure $\mu(X)$ may be attributed
- The utility of $y \in Y$ is $u(y, X)$
- There are N opinions given by probability density functions $p_{M_1}(X), \ldots, p_{M_N}(X)$

Thus, for a probability density function $p(X)$,

$$u(y|p(X)) = \int u(y, X)p(X)d\mu(X) \tag{2.1}$$

The rule for choosing y is stated as follows: " Choose weights $w_1, \ldots, w_N$ ($w_i \geq 0, i = 1, \ldots, N$, and $\sum_{i=1}^{N} w_i = 1$); construct the pooled density function $p_{M_\lambda}(X) = \sum_{i=1}^{N} w_i p_{M_i}(X)$; choose the y maximizing $u(y|p_{M_\lambda}(X))$" This rule is called *Linear Opinion Pool*. The rule can be made democratic by setting $w_1 = \cdots = w_N = \frac{1}{N}$.

Let $p_a(X)$ denote the actual, operative probability distribution. It is proved by Stone that:

- If, for some μ_1, μ_2, $p_a(X) = \mu_1 p_{M_1}(X) + \mu_2 p_{M_2}(X)$, then, $u(y_{M_\lambda}|p_a(X)) \geq \min_{i=1,\ldots,N} u(y_{M_i}|p_a(X))$ holds for any weights w_1, w_2. (It is assumed that y_{M_1}, y_{M_2}, y_{M_λ} exist.)

- If

  1. Y is an interval of real numbers

  2. $u(y, X)$ is, for each X, a strictly convex function of y

  then, $u(y_{M_\lambda}|p_a(X)) \geq \min_{i=1,\ldots,N} u(y_{M_i}|p_a(X))$ holds for all weights w_1, ..., w_N. (It is assumed that y_{M_1}, ..., y_{M_N}, y_{M_λ} exist.)

By adopting it into multimedia fusion, the decisions or posteriors from each information source are combined linearly [Punska, 1999]. Let $\mathbf{X}^{(i)}$ be the observations from the source M_i, and N be the total number of information sources. It is defined as:

$$p(\mathbf{y}|\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \ldots, \mathbf{X}^{(N)}) \propto \sum_{i=1}^{N} w_i p(\mathbf{y}|\mathbf{X}^{(i)}) \tag{2.2}$$

where w_i is a weight such that, $0 \leq w_i \leq 1$ and $\sum_{i=1}^{N} w_i = 1$. The weight w_i reflects the significance attached to the source M_i. In literature, there are various methods for weight normalization such as min-max, decimal scaling, z-score, tanh-estimators and sigmoid function. The Linear Opinion Pool is illustrated in Figure 2.1.

A variation of Linear Opinion Pool is *Logarithmic Opinion Pool* [Heskes, 1998]. If the output of a network is interpreted as a probability statement, the sum-squared

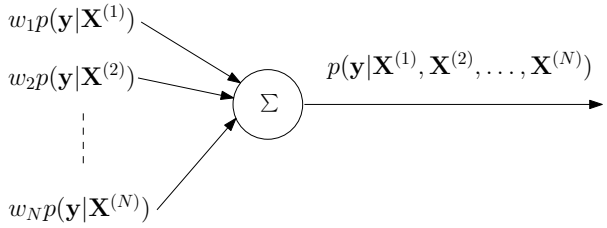


Figure 2.1: Linear Opinion Pool

error corresponds to the negative of its log likelihood or equivalently the Kullback-Leibler divergence, and linear averaging of the outputs corresponds to logarithmic averaging of the probability statements: Logarithmic Opinion Pool.

The distance between the true probability $q(y|X)$ and an estimated $p(y|X)$ is modeled as the Kullback-Leibler divergence:

$$K(q,p) = -\int dX p(X) \int dy q(y|X) \log[\frac{p(y|X)}{q(y|X)}] \tag{2.3}$$

The average model $p(y|X)$ is defined to be the one that is closest to the given set of models:

$$p(y|X) = \arg \min_{p(y|X)} \sum_i w_i K(p, p_i) \tag{2.4}$$

Introducing a Lagrange multiplier for the constraint $\int dX p(y|X) = 1$, the solution is:

$$p(y|X) = \frac{1}{Z(X)} \prod_i [p_i(y|X)]^{w_i} \tag{2.5}$$

with normalization constant $Z(X) = \int dy \prod_i [p_i(y|X)]^{w_i}$. It can be written for multimedia fusion as:

$$p(\mathbf{y}|\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \ldots, \mathbf{X}^{(N)}) = \alpha \prod_i [p(\mathbf{y}|\mathbf{X}^{(i)})]^{w_i} \tag{2.6}$$

and it is illustrated in Figure 2.2.

The Logarithmic Opinion Pool is "externally Bayesian", *i.e.*, can be derived from joint probabilities using Bayes' rule [Bordley, 1982]. But the complete pool assigns probability zero if any source assigns zero. The main problem for both Linear Opinion Pool and Logarithmic Opinion Pool is how to choose the weights w_i [Heskes, 1998].

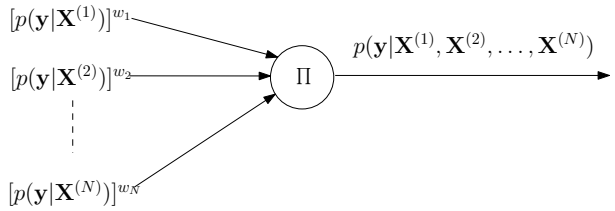


Figure 2.2: Logarithmic Opinion Pool

2.1.1.2 Independent Opinion Pool

By assuming the information obtained conditioned on the observation set $p(\mathbf{y}|\mathbf{X}^{(m)})$ is independent, the *Independent Opinion Pool* is derived in [Manyika and Durrant-Whyte, 1994]:

$$p(\mathbf{y}|\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \ldots, \mathbf{X}^{(N)}) = \alpha \prod_{i=1}^{N} p(\mathbf{y}|\mathbf{X}^{(i)}) \tag{2.7}$$

where α is a normalizing constant. The Independent Opinion Pool is defined by the following equation:

$$p(\mathbf{y}|\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \ldots, \mathbf{X}^{(N)}) \propto \prod_{i=1}^{N} p(\mathbf{y}|\mathbf{X}^{(i)}) \tag{2.8}$$

and is illustrated in Figure 2.3. In general, the independence assumption is difficult

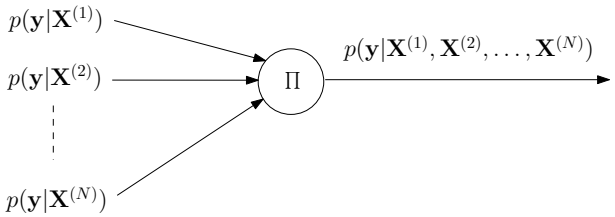


Figure 2.3: Independent Opinion Pool

to satisfy. However, in the the realm of measurement and experimentation based on physical laws and principles, the conditional independence can often be justified experimentally [Manyika and Durrant-Whyte, 1994]. This is usually done by showing that the residual uncertainty in each observation arises from the uncorrelated noise terms.

One drawback of the Independent Opinion Pool is that it needs to have prior

information about the probability distribution of each source.

$$p(\mathbf{y}|\mathbf{X}^{(1)},\mathbf{X}^{(2)},\ldots,\mathbf{X}^{(N)}) = \alpha[\frac{p_{M_1}(\mathbf{X}^{(1)}|\mathbf{y})p_{M_1}(\mathbf{y})}{p_{M_1}(\mathbf{X}^{(1)})} \times \cdots \times \frac{p_{M_N}(\mathbf{X}^{(N)}|\mathbf{y})p_{M_N}(\mathbf{y})}{p_{M_N}(\mathbf{X}^{(N)})}] \quad (2.9)$$

If the prior information at each information source is common, *i.e.*, obtained from the same source, then $p_{M_1}(\mathbf{y}) = \cdots = p_{M_N}(\mathbf{y})$ and this results in unwanted extreme reinforcement of opinion. The Independent Opinion Pool is only appropriate when the priors are obtained independently on the basis of subjective prior information at each information source.

A variation of Independent Opinion Pool is *Independent Likelihood Pool* when each information source has common prior information, *i.e.*, information obtained from the same origin.

$$p(\mathbf{y}|\mathbf{X}^{(1)},\mathbf{X}^{(2)},\ldots,\mathbf{X}^{(N)}) = \frac{p(\mathbf{X}^{(1)},\mathbf{X}^{(2)},\ldots,\mathbf{X}^{(N)}|\mathbf{y})p(\mathbf{y})}{p(\mathbf{X}^{(1)},\mathbf{X}^{(2)},\ldots,\mathbf{X}^{(N)})} \quad (2.10)$$

By assuming that the likelihoods from each information source are independent, the Equation 2.10 can be written as:

$$\begin{aligned} p(\mathbf{y}|\mathbf{X}^{(1)},\mathbf{X}^{(2)},\ldots,\mathbf{X}^{(N)}) &= \frac{p(\mathbf{y})p(\mathbf{X}^{(1)}|\mathbf{y})p(\mathbf{X}^{(2)}|\mathbf{y})\cdots p(\mathbf{X}^{(N)}|\mathbf{y})}{p(\mathbf{X}^{(1)},\mathbf{X}^{(2)},\ldots,\mathbf{X}^{(N)})} \\ &= \frac{p(\mathbf{y})\prod_{i=1}^{N} p(\mathbf{X}^{(i)}|\mathbf{y})}{p(\mathbf{X}^{(1)},\mathbf{X}^{(2)},\ldots,\mathbf{X}^{(N)})} \\ &= \alpha p(\mathbf{y})\prod_{i=1}^{N} p(\mathbf{X}^{(i)}|\mathbf{y}) \end{aligned} \quad (2.11)$$

where α is a normalizing constant. The Independent Likelihood Pool is illustrated in Figure 2.4.

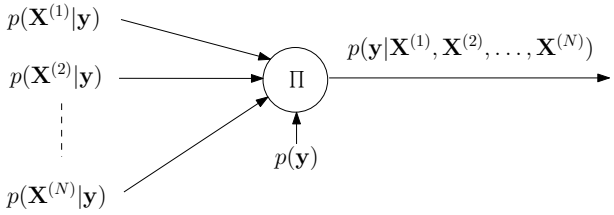


Figure 2.4: Independent Likelihood Pool

2.1.1.3 Maximum, Minimum and Median Rules

The Maximum fusion takes the maximum prediction score of all information sources as the final prediction score. The maximum rule can be expressed as:

$$I(\mathbf{X}) = \max_i I_i(\mathbf{X}) \tag{2.12}$$

However, this immediately fails if some classifiers are more overtrained than others [Duin, 2002]. In that case they may be overconfident and thereby dominate the outcome, without having a better performance. The maximum rule also fails for simple classifiers that are not sensitive for nuances that more complicated, and thereby better, classifiers are able to detect [Duin, 2002].

The Minimum fusion takes the minimum prediction score of all information sources as the final prediction score. The minimum rule can be expressed as:

$$I(\mathbf{X}) = \min_i I_i(\mathbf{X}) \tag{2.13}$$

Like for the maximum rule, a good example of a situation in which this rule is really adequate is hard to find [Duin, 2002].

The Median fusion takes the median prediction score of all information sources as the final prediction score. The median rule can be expressed as:

$$I(\mathbf{X}) = median_i(I_i(\mathbf{X})) \tag{2.14}$$

In general, these rules can be considered as special cases of linear fusion. By assigning the weight of the maximum prediction as 1, it is Maximum fusion. By assigning the weight of the minimum prediction as 1, it is Minimum fusion. By assigning the weight of the median prediction as 1, it is Median fusion.

2.1.1.4 Majority Voting

In *majority voting* based fusion, the final decision is the one where the majority of the information sources reach a similar decision [Sanderson and Paliwal, 2004]. By

introducing weights on different information sources, we have the *weighted majority voting* which is used in Adaboost [Freund and Schapire, 1997]. The majority voting is a special case of weighted combination with all weights to be equal.

The classifier selection problem for majority voting is studied in [Ruta and Gabrys, 2005] on multi-classifier system. The authors compared different selection criterions experimentally to combine different classifiers obtained with different algorithms on the same dataset: Mean classifier error, Majority voting error, Product-moment correlation, Double-fault measure, Q statistics measure, *etc.*. The better the correlation between the measure (selection criterion) and the combiner performance, the higher the performance of the selected combinations. Ultimately, majority voting error used as a selection criterion showed the optimal results.

2.1.1.5 Dempster-Shafer Theory

The *Dempster-Shafer Theory* (DST) [Shafer, 1976] is an effective tool for combining measures of evidence. It is well-known for its usefulness to express uncertain judgements of experts and is used in many data fusion applications such as [Braun, 2000; Koks and Challa, 2003] based on two ideas: obtaining degrees of belief for one question from subjective probabilities for a related question, and Dempster's rule for combining such degrees of belief when they are based on independent items of evidence [Shafer, 1992]. The advantage of DST is that it allows coping with absence of preference, due to limitations of the available information, which results in indeterminacy. The theory is often viewed as a generalization of Bayesian probability theory, by providing a coherent representation for *ignorance* (lack of evidence) and also by discarding the *insufficient reasoning principle*. However, the two approaches (Bayesian and Dempster-Shafer Theory) differ significantly and the extent of their applicability to data fusion is still being debated [Braun, 2000]. Bayesian theory is based on the classical ideas of probability, while Dempster-Shafer Theory is a recent attempt to allow more interpretation of what

uncertainty is all about [Koks and Challa, 2003]. DST contains two new ideas that are foreign to Bayesian theory. These are the notions of support (belief) and plausibility [Koks and Challa, 2003]. Many researchers have been inspired to investigate different aspects related to uncertainty or imprecision and lack of knowledge and their applications to real life problem. The DST covers several different models, such as the theory of hints [Kohlas and Monney, 1995] and the transferable belief model (TBM) [Smets, 1998].

Some fundamentals of the Dempster-Shafer theory are first introduced here. In DST, which is also referred to as *evidence theory* or the *Dempster-Shafer Evidential Theory*, evidence is represented in terms of *evidential functions* and *ignorance*. These functions include *mass functions* (or *basic probability assignment function*, m), *belief functions* (bel), and *plausibility functions* (pl) [Shafer, 1976].

Let Θ be a finite non-empty set, called the *frame of discernment* [Bi, Guan, and Bell, 2008]. The elements of Θ are the hypotheses. Its power set 2^{Θ} is the set of all subsets. A mass function is a mapping function $m : 2^{\Theta} \rightarrow [0, 1]$ such that:

$$\begin{aligned} m(\phi) &= 0 \\ \sum_{X \subseteq \Theta} m(X) &= 1 \end{aligned} \tag{2.15}$$

ϕ is the null set. The first property requires an appropriate choice of the universal set Θ. That means, the set Θ has to be complete and contain all possible hypotheses of the scenario considered. The second property means that all statements of a single data source have to be normalized, just to ensure that the evidence presented by each data source is equal in weight, *e.g.* no data source is more important than another one [Kay, 2007]. A mass function is a *basic probability assignment*(BPA) to all subsets X of Θ. A subset A of a frame Θ is called a *focal element* or *focus* of a mass function m over Θ if $m(A) > 0$ and A is called a *singleton* if it is a one-element subset.

A function $bel : 2^{\Theta} \rightarrow [0,1]$ is called a *belief function* if it satisfies:

$$\begin{aligned} bel(\phi) &= 0 \\ bel(\Theta) &= 1 \\ bel(A_1 \bigcup A_2 \bigcup \cdots \bigcup A_n) &= \sum_{1 \leq i \leq n} bel(A_i) - \\ &\quad \sum_{1 \leq i \leq j \leq n} bel(A_i \bigcap A_j) + \cdots \\ &\quad + (-1)^{n+1} bel(A_1 \bigcap A_2 \bigcap \cdots \bigcap A_n) \end{aligned} \tag{2.16}$$

Here, $A_1, A_2, \ldots, A_n$ is subsets of Θ. The measure represents the total evidence or belief that the element belongs to the set as well to its various special subsets, so $bel(A) = \sum_{B \subseteq A} m(B)$ [Telmoudi and Chakhar, 2004].

A function $pl : 2^{\Theta} \rightarrow [0,1]$ is called a *plausibility function* if it satisfies [Telmoudi and Chakhar, 2004]:

$$\begin{aligned} pl(A_1 \bigcap A_2 \bigcap \cdots \bigcap A_n) &= \sum_{1 \leq i \leq n} bel(A_i) - \\ &\quad \sum_{1 \leq i \leq j \leq n} bel(A_i \bigcup A_j) + \cdots \\ &\quad + (-1)^{n+1} bel(A_1 \bigcup A_2 \bigcup \cdots \bigcup A_n) \end{aligned} \tag{2.17}$$

The plausibility function represents not only the total evidence or belief that the element in question belongs to the set or to any of its subsets but also the additional evidence or belief associated with sets that overlap with it, so $pl(A) = 1 - bel(\overline{A}) = \sum_{B \bigcap A \neq \phi} m(B)$ [Telmoudi and Chakhar, 2004].

For a set A, the *belief function* and *plausibility function* are illustrated as in Figure 2.5. The belief is a kind of loose lower limit to the uncertainty. On the other hand, the plausibility is a loose upper limit to the uncertainty.

The combination of evidence from different sources is accomplished within the basic DST formalism by the Dempster combination rule:

$$m_c(H) = \frac{\sum_{X \bigcap Y = H} m_1(X) m_2(Y)}{1 - \sum_{X \bigcap Y = \phi} m_1(X) m_2(Y)} \tag{2.18}$$

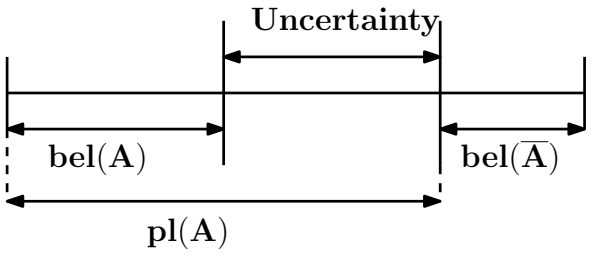


Figure 2.5: The illustration of the belief and plausibility

where m_c is the combined BPA for a given hypothesis H.

Dempster-Shafer Theory and Bayesian methods both offer mechanisms with which some of the sensor fusion fundamental problems of information uncertainty, conflicts, incompleteness, and disparity, can be approached [Braun, 2000]. In the field of sensor data fusion, the respective advantages of the two formalisms remain a topic of interest [Braun, 2000]. The results of the simulations in [Braun, 2000] show that both methods are robust over the entire sensor information domain, and generally where one succeeds or fails the other will do the same, with just a slight edge being given to Dempster-Shafer as compared with the Bayesian approach [Koks and Challa, 2003]. The Dempster-Shafer method has several other advantages over Bayesian decision theory. Most importantly, hypotheses do not have to be mutually exclusive, and the probabilities involved can be either empirical or subjective [Dailey, Harn, and Lin, 1996]. DST also has some shortcomings. As one might expect, application of the Dempster-Shafer method demands extensive computational capabilities [Dailey, Harn, and Lin, 1996]. The calculations tend to be longer and more involved than their Bayes analogues (which are not required to work with all the elements of a set) [Koks and Challa, 2003]. Other shortcomings of DST, include the manner in which it handles conflicting information and its reliance on the basic assumption that two pieces of evidence must have the same population universe [Dailey, Harn, and Lin, 1996]. Despite the fact that reports such as [Cremer, den Breejen, and Schutte, 1998] and [Braun, 2000] indicate that Dempster-Shafer can sometimes perform better than Bayes theory, Dempster-Shafer's

computational disadvantages do nothing to increase its popularity [Koks and Challa, 2003].

2.1.2 Representative works of rule-based fusion

Rule-based fusion strategies have been adopted for performing various multimedia analysis tasks. Some representative works will be discussed in the following chronologically.

Neti *et al.* [Neti et al., 2000] presented a unified framework for fusion of audio and visual information for speech recognition, speaker recognition, speaker change detection and speech event detection. In the speech recognition problem, individual numerical scores obtained using audio and visual features are fused in a weighted product way. This product fusion assumes that the two streams of information are independent, especially when individual scores are interpreted as probabilities of occurrences of the symbolic units associated with the two streams. In practice such an independence assumption could be debated, especially since the two streams are realizations of the same perceptual process synchronously observed in time. In the speaker recognition problem, the average similarity over the facial features between the test candidate and the face template is computed as the visual similarity. The sum of the distance over all the test frames is used as the audio likelihood. Here, the distance is the logarithm of the likelihood between test frame and the speaker *Gaussian Mixture Model*(GMM). Scores are fused using a linear weighted sum: the weight for visual similarity is $\cos\alpha$ and the weight for audio similarity is $\sin\alpha$, where α is selected according to the relative reliability of audio and face identification. In speaker change detection, the difference between the *Bayesian information criterion*(BIC) values is considered as the audio information score. The visual score is computed using Kullbach-Liebler type divergence criterion. Then, scores are fused using a linear weighted sum. In speech event detection, two probability densities are computed using audio and visual features respectively as GMM. A simple linear weighted sum fusion strategy is then used.

Lucey *et al.* in [Lucey, Sridharan, and Chandran, 2001] used audio and visual modalities fusion for speech recognition. At first, the individual word likelihood scores are obtained through *Hidden Markov Model*(HMM) using audio or visual modalities. Then, essentially a weighted fusion rule is applied on the scores. The weights for different modalities are sum up to 1. By considering the likelihoods as a feature vector, the secondary classification method measures the correctness of audio and video modalities on word basis and thus adaptively weight the audio and visual modalities. Bhattacharyya distance is used to measure the separability of correct and incorrect likelihoods distributions. If only one modality is classified as correct, the weight of the modality is set to 1. If both are correct, the weights are set to 0.5.

Foresti *et al.* presented a sensor fusion method for tracking in [Foresti and Snidaro, 2002]. The tracking procedure fuses information coming from the different sensors in the distributed sensor network. Each sensor is associated with a dynamically changing reliability factor as a confidence measure. The trajectory coordinates obtained from each sensor are weighted averaged in order to estimate the correct location of the blob. The weights are estimated using appearance ratio in [Snidaro et al., 2004].

Iyengar *et al.* combined visual and audio information for the monologue detection in [Iyengar, Nock, and Neti, 2003a]. The monologue is defined as "detection of video segments which contain(s) an event in which a single person is at least partially visible and speaks for a long time without interruption by another speaker." According to the definition, the monologue is related to both visual and audio steams. The method utilizes face scores, speech scores, and synchrony scores.

- The face score is calculated as the ratio between the likelihood based on GMMs trained on frontal face and non-face images.
- The speech score is the normalized concept score obtained through the audio concept HMMs.

- The mutual information between each pixel in the video frames and the audio features are computed. The synchrony score is derived as the ratio between the best mutual information region and the background.

The three scores are fused using linear weighted sum and weighted product with the independence assumption. Here, the weight for each information source is obtained using grid search in the range $(0, 1)$.

Polikar *et al.* introduced a supervised incremental learning algorithm *Learn++* in [Polikar et al., 2001]. The algorithm is based on generating a number of hypotheses using different distributions of the training data and combining these hypotheses using a weighted majority voting. Later on, Parikh *et al.* applied Learn++ to the multimodal fusion problem [Parikh et al., 2004]. The classifier is trained from each modality data in the way similar to Adaboost. Then, the different classifiers from different modality datasets are combined together using weighted majority voting. It is illustrated in Figure 2.6. In Figure 2.6(a), $M_1, ..., M_N$ are the multimodal data. Hypotheses $H_1, ..., H_N$

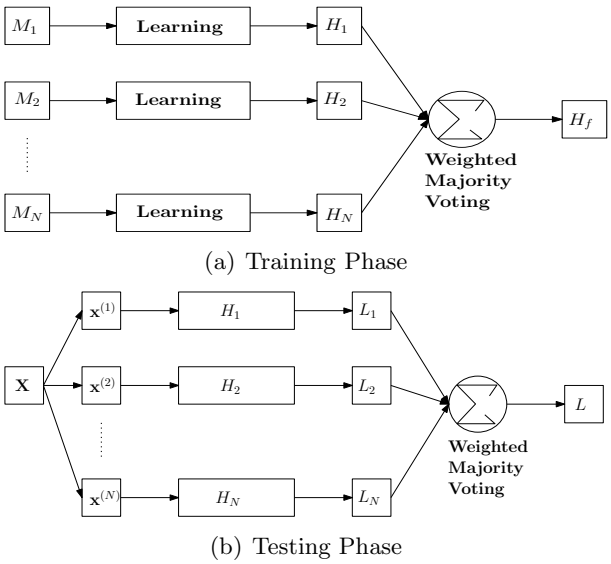


Figure 2.6: The illustration of the Learn++ Fusion

are obtained through the learning algorithm for each modality data. The hypotheses

are combined using weighted majority voting to develop the final hypothesis H_f. In Figure 2.6(b), different modalities $\mathbf{X}^{(1)}, ..., \mathbf{X}^{(N)}$ of testing data X are tested using corresponding hypothesis. The final label L of X is obtained by combining the results from each modality.

Atrey *et al.* [Atrey, Kankanhalli, and Jain, 2006] presented a novel framework for multiple sensor fusion to detect events in the surveillance and monitoring systems. The decisions of media streams are sequentially combined using the Logarithmic Opinion Pool while considering the confidence and correlation. The Bayesian method is used to fuse the confidence levels in individual streams. "Agreement coefficient" is introduced to measure the correlation among the media streams. The agreement fusion model is based on average-link clustering. Recently, the authors proposed a confidence evolution method in [Atrey and Saddik, 2008]. The method dynamically computes the confidence level of different modalities based on the past history of their agreement with the trusted streams. The limitation is that the trusted streams whose confidence is above a certain threshold need to be specified first.

Yang *et al.* applied multimedia fusion method to web video categorization in [Yang et al., 2007]. Web videos have rich information from multiple information sources. Not only visual and audio information, but also the surrounding text (the titles, descriptions and tags of web videos), and even the social (*i.e.* the relationship among videos through the users or the recommendations) information can be applied. Visual low-level features, semantic features (*e.g.*, concept histogram), audio feature, and surrounding text features are extracted. The scores for each information source are obtained using *Support Vector Machine* (SVM) and then fused using max fusion, average fusion, and linear weighted fusion (the weight is selected according to the average precision of each single modality), respectively. It is shown that the linear weighted fusion outperforms the other two fusion methods in this application.

A probabilistic fusion method is proposed in [Zheng et al., 2008]. Relevance

Vector Machine (RVM) [Tipping, 1999] is used to train classifiers for different information sources to obtain the probabilistic outputs. Then, the outputs are fused using Independent Opinion Pool for concept detection.

A method for enhancing the performance of a correlated biometrics verification system is presented in [Srinivas, Veeramachaneni, and Osadciw, 2009]. The outputs of correlated biometric classifiers are dynamically weighted. The particle swarm optimization (PSO) algorithm which updates the weights in each iteration is applied to a training dataset to obtain the weights that minimizes the Bayesian risk function. The correlation among different information sources is not explicitly considered.

A multimodal fusion method exploiting complementary information stemming from multiple information sources is proposed in [Papandreou et al., 2009] to improve performance by uncertainty compensation. The authors adopted the product of each feature probability as the fusion rule. The adaptive compensation is considered to account for the observation uncertainty. The observation noise for each information source is considered Gaussian, and the feature probability is modeled using GMM. Thus, the noisy observations are compensated by shifting the models means and increasing the model covariances. This method is demonstrated in the application of audio-visual automatic speech recognition. However, reliable methods for dynamically estimating the feature observation uncertainty are needed in the method. Moreover, the noise is approximated using a Gaussian model which may not always be true.

The previous methods are generally static, *i.e.*, the fusion function for the entire data. Geng *et al.* [Geng et al., 2010] proposed a context-aware fusion method. The fusion method uses linear weighted sum fusion rule. But instead of static fusion weights, the method introduced context factors to dynamically adapt the weights to the environment. For example, in the human identification problem, gait and face information is fused. But the context factors, *i.e.*, view angle and subject-to-camera distance, may effect the reliability relationship between gait and face. The effect of context factors to

the relationship between different information sources are based on either prior knowledge or machine learning. For the prior knowledge based method, more parameters need to be empirically determined and carefully tuned to suit different dataset. For the machine learning (neural network) based method, neural network can learn non-linear relationship, but it is difficult to scale and must be re-trained when the dataset changes. It is illustrated in Figure 2.7.

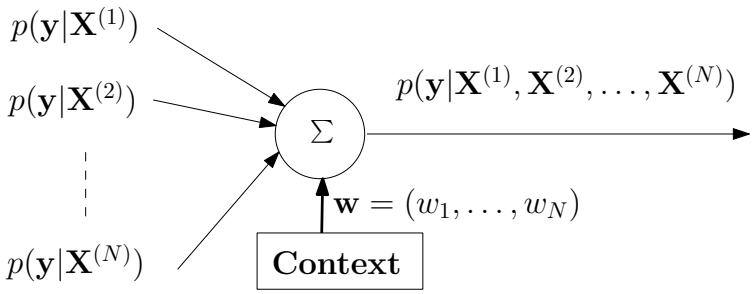


Figure 2.7: Context-aware linear fusion

A probabilistic framework that combines *Context* and *Content* for processing video information is introduced in [Jasinschi et al., 2002]. The *Context* and *Content* are represented in two layers using Bayesian networks. Hierarchical priors provide the connection between the two layers. The integration of content and context information is formalized using Chapman-Kolmogorov equation. The classification task can be improved by combining context and content information. However, prior knowledge between the different features is needed and it is difficult to generalize the method for other applications.

In the following part, the works using Dempster-Shafer Theory (DST) are discussed chronologically.

Transferable belief model is used in multimedia fusion in [Guironnet, Pellerin, and Rombaut, 2005] for video concept detection. The video is first segmented into shots. One or more keyframes are then extracted from the shot. The color features and texture features are extracted and principal component analysis (PCA) is performed

to reduce feature dimensionality. Then, Support Vector Machine (SVM) learning is used to recognize a given concept individually. The fuzzy sets can be used to model the Basic Belief Assignment (BBA) from the output of the SVM learned on a given concept. The set of hypotheses is defined $\Omega = \{H, \overline{H}\}$ where H is a given concept and $\overline{H}$ is not a given concept. The sensor fusion of color and texture is performed for each concept to improve the classification. A mass is often assigned to the empty set known as conflict, and the mass in conflict is transferred to avoid making decision. The authors also suggested to use the concept fusion modeling interaction between concepts if there is a relation existing between the two concepts to improve the performance.

Singh *et al.* [Singh et al., 2006] presented a fingerprint classification fusion algorithm using Dempster-Shafer theory. The authors used DST to combine the decisions of three different fingerprint classification algorithms based on the minutiae, ridge and fingercode, respectively. The belief function associated with each algorithm is revised using the Dempster's update rule when new evidences are added. The fusion method outperforms many other fusion algorithms such as sum rule and min-max rule.

Reddy *et al.* [Reddy, 2007] also used the DST for fusing the outputs of two sensors, the hand gesture recognizer and the Brain Computing Interface. It is shown that the fusion system is able to resolve the ambiguity between the concepts satisfactorily under various operating scenarios.

Bi *et al.* [Bi, Guan, and Bell, 2008] proposed a "class-indifferent" method for combining classifier decisions represented by evidential structures called *triplet* and *quartet*, using Dempster's rule of combination. The authors presented a formalism for modeling classifier decisions as triplet mass functions and established a range of formulae for combining these mass functions in order to arrive at a consensus decision. The comparison made between Dempster's rule and majority voting over the UCI benchmark data showed that DST is better than majority voting in combining the individual classifiers.

2.1.3 Summary

The weighted-sum strategy is more tolerant to noise because sum does not magnify noise as severely as the product [Wu et al., 2004]. In comparison, the Independent Opinion Pool is highly sensitive to noise [Wu et al., 2004]. The work of Tax *et al.* [Tax et al., 2000] concluded that the product-combination rule works well only when the posterior probability of individual classifiers can be accurately estimated. If there are dependencies between information sources, the Linear Opinion Pool should be used instead of the Independent Opinion Pool [Punska, 1999]. In addition to the fact that we will not have truly independent modalities, we generally cannot estimate posterior probabilities with high accuracy [Wu et al., 2004].

There are also some experimental studies in Multi-Classifier Systems (MCSs). In [Kittler et al., 1998], the various fusion schemes, such as the product rule, sum rule, minimum rule, maximum rule, median rule, and majority voting were compared experimentally. It is shown that the sum rule outperforms other fusion schemes since the sum rule is most resilient to estimation errors.

It is concluded in [Tax et al., 2000] averaging-estimated posterior probabilities is to be preferred in the case when posterior probabilities are not well estimated. Only in the case of problems involving multiple classes with good estimates of posterior class probabilities the product combination rule outperforms the mean combination rule. Alexandre *et al.* [Alexandre, Campilho, and Kamel, 2001] compared two types of averaging combination rules: arithmetic mean (equal weights for Linear Opinion Pool) and geometric mean (equal weights for Logarithmic Opinion Pool). For a problem with two classes, these rules have exactly the same performance when using two classifiers, while geometric mean performs better when more than two classifiers are combined.

The Dempster-Shafer Theory (DST) is an effective tool for combining multiple evidence. The main advantage of DST is that no a priori knowledge is required. More-

over, a value for ignorance can be expressed, given information on the uncertainty of a situation. It has been found more suitable for handling mutually inclusive hypotheses. However, the *mass functions* and *belief functions* need to be designed for different applications, which degrades the portability of DST. It also suffers from the combinatorial explosion and conflicting beliefs problem. A belief function must distribute belief to the power set of the universal set (*frames of discernment*). Thus, the computational complexity increases exponentially with the number of *frames of discernment* Θ [Chen and Aickelin, 2006]. Dempster's rule of combination redistributes the mass values of empty propositions to non-empty propositions, also known as normalization step, due to the definition of the mass function. This sometimes leads to erroneous results, which causes the conflicting management problem [Chen and Aickelin, 2006].

A summary of all the linear fusion works described above is provided in Table 2.1. The linear fusion model is easy to adopt and does not need much computation. It is easy to scale. A theoretical framework for bounding the average precision of a linear combination function in video retrieval is presented in [Yan and Hauptmann, 2003]. The authors concluded that the linear combination functions have limitations, and suggested that non-linearity and cross-media relationships should be introduced to achieve better performance. Moreover, the correlations among different information sources in fusion is not well studied. The determination of optimal weights for different information sources is still an open problem.

Table 2.1: A list of the representative works in linear fusion methods

Work	**Fusion method**	**Weight type**	**Correlation**	**Uncertainty**	**Multimedia analysis task**
[Neti et al., 2000]	linear weighted sum with relative reliability for weights	static	Not considered	Not considered	speaker recognition, speaker change detection and speech event detection using visual and audio information
Lucey *et al.* [Lucey, Sridharan, and Chandran, 2001]	weighted product with confidence measured by secondary classification for weights	word based	Not considered	Not considered	speech recognition
[Foresti and Snidaro, 2002]	linear weighted sum with appearance ratio for weights	dynamic	Not considered	Not considered	tracking

Table 2.1 – continued from previous page

Work	**Fusion method**	**Weight type**	**Correlation**	**Uncertainty**	**Multimedia analysis task**
Iyengar *et al.* [Iyengar, Nock, and Neti, 2003a]	linear weighted sum and weighted products with grid search for weights	static	Not considered	Not considered	monologue detection using visual and audio information
Parikh *et al.* [Parikh et al., 2004]	weighted majority voting with weights related to accuracy	static	Not considered	Not considered	identifying defects in pipelines
[Wu and McClean, 2005]	linear weighted sum with weights related to average correlation coefficient	static	Simply considered	Not considered	document retrieval
Atrey *et al.* [Atrey, Kankanhalli, and Jain, 2006]	logarithmic opinion pool using agreement coefficient for correlation	static	Simply considered	Not considered	event detection
Yang *et al.* [Yang et al., 2007]	linear weighted sum with average precision for weights	static	Not considered	Not considered	web video categorization

Table 2.1 – continued from previous page

Work	**Fusion method**	**Weight type**	**Correlation**	**Uncertainty**	**Multimedia analysis task**
Atrey *et al.* [Atrey and Saddik, 2008]	logarithmic opinion pool using agreement coefficient for correlation	dynamic	Simply considered	Not considered	event detection
[Zheng et al., 2008]	product rule	static	Not considered	Not considered	concepts detection
[Srinivas, Veeramachaneni, and Osadciw, 2009]	linear weighted sum with particle swarm optimization for weights	static	Not considered	Not considered	biometrics verification
[Papandreou et al., 2009]	weighted product with weights related to Gaussian noise model	static	Not considered	Not considered	speech recognition
[Geng et al., 2010]	linear weighted sum with weights related to context factors	dynamic	Not considered	Not considered	human identification

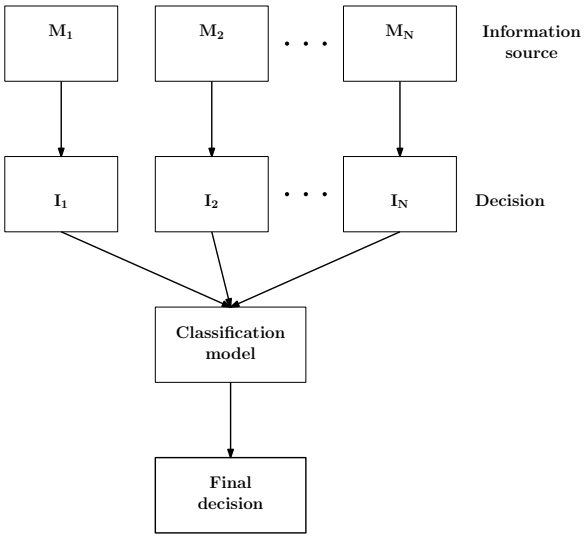


Figure 2.8: The illustration of the classification-based fusion

2.2 Classification-based Fusion

Besides the rule-based fusion models, some researchers have also sought to use classification-based methods for multimedia fusion. With the various powerful classification algorithms, effective fusion results can be obtained.

2.2.1 Overview of methods

The general procedure for classification-based fusion are as follows: With the multimedia data, the decisions can be obtained from each information source after certain processing. Then, the decisions from different information sources are composed and passed to classification algorithms as input. The output of the classification model is taken as the fused decision. It can be illustrated as in Figure 2.8.

2.2.2 Representative works

The representative classification-based decision fusion methods are discussed in the following chronologically.

Adams *et al.* [Adams et al., 2003] adopted SVM based decision fusion for

semantic-concepts detection (*e.g.*, sky, water, speech) using visual, audio, and text information. The semantic-concept is modeled as a class conditional probability density function over a feature space. The authors used GMMs for independent observation vectors and HMMs for time series data. Initially, concept models in the individual information sources are learned to generate the scores for concept. Then, the concept scores for individual information sources are integrated into a vector and passed to SVM. In this way, the fusion model can be learned and the classification results of score vectors are considered as the fusion results. A similar approach has been presented in [Iyengar, Nock, and Neti, 2003b].

Wu *et al.* proposed a two-step multimodal fusion approach for multimodal data analysis in [Wu et al., 2004]. The authors pointed out that there are three design factors that affect fusion performance: *modality independence*, *curse of dimensionality*, and *fusion-model complexity*. The two-step approach achieves a careful balance of the three design factors. In the first step, statistically independent modalities from raw features are extracted using independent modality analysis (IMA). The procedure consists of the following three stages: principal component analysis (PCA), independent component analysis (ICA), and independent modality grouping (IMG). PCA technique is first applied to raw features to remove noise and reduce the feature dimensionality. The first k principal components are obtained as the output. Then, by assuming that the observations are mixture signals coming from k unknown independent components, ICA is performed on the main eigenvectors of PCA representations to determine which PC's actually are independent and which should be grouped together as parts of a multidimensional component. Since the resulting k components from ICA might not be independent, and the number of components can be too large to face the challenge of "dimensionality curse". IMG is adopted to divide k components into D groups by minimizing inter-group feature correlation and maximizing intra-group feature correlation. Here, the value of D is experimentally determined. In the second step, *super-kernel*

fusion is used to determine the optimal combination of individual modalities. Given D independent modalities, individual classifiers are trained and the posterior probability for each training sample is estimated to obtain a super-kernel matrix. The scores are again trained using SVM to yield the fusion model. For a new test sample which is represented in raw features, it is divided into D modalities. Then, posterior probability for each modality is calculated and formed into a vector of D elements. The vector is then inputted into the fusion model to achieve a class prediction. Independent modality analysis can improve the effectiveness of multimedia data analysis by achieving a tradeoff between dimensionality curse and modality independency. On the other hand, super-kernel fusion has high model complexity and can explore interdependencies between modalities.

Zhu *et al.* [Zhu, Yeh, and Cheng, 2006] reported a multimodal fusion method for image categorization by combining visual and text cues. For visual cue, the top K categories are selected based on the bag-of-words model. For text cue, the text line is detected and described in a 16-Dimension feature. The text concept is learned using Multiple Instance Learning (MIL). Then, K probabilities are obtained using visual cue and the weighted Euclidian distance are obtained using text cue. The features calculated from these information are assembled as the input for SVM-based classification. Here, pair-wise binary SVM classifiers (PWC) using a linear kernel are adopted to fuse the visual and text cues. The PWC needs $\frac{K\times(K-1)}{2}$ classifications. The computation cost becomes the major limitation as K increases.

An Integrated Statistical Model (ISM) is proposed in [Gao, Lim, and Sun, 2007]. Different from the traditional fusion models where only the original value was used, the ISM fusion model used the deep structure of modality distribution. The modality may contain rich structures. Each structure is modeled by a Gaussian distribution which is called the mode. Each modality will have K modes to characterize its distribution. Given an observed modality value, the mode with the maximal probability is chosen as

the corresponding mode identity. For all the modalities of one object, the corresponding mode configuration is treated as a document. The co-occurrence mode features similar to that adopted in text categorization [Nigam, Lafferty, and McCallum, 1999] is extracted. The concept models are then trained using maximum entropy (ME) approach [Berger, Pietra, and Pietra, 1996]. Then, the predict concept probability for a concept is calculated as the exponential of weighted features, and the concept with the maximal probability is assigned to the document.

Li *et al.* proposed an ordered weighted aggregation (OWA) based fusion method in [Li et al., 2009]. The ordered weighted aggregation operator was first introduced by Yager [Yager, 1988]. A mapping F from $[0,1]^n \rightarrow [0,1]$ is called an OWA operator of dimension n if associated with F is a weighting vector $\mathbf{W} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix}$ such that

1. $w_i \in (0,1)$
2. $\sum_i w_i = 1$

and where

$$F(a_1, a_2, \ldots, a_n) = w_1 b_1 + w_2 b_2 + \cdots + w_n b_n \tag{2.19}$$

where b_i is the ith largest element in the collection $a_1, a_2, \ldots, a_n$. It is proved that OWA operators are monotonic with respect to argument values. Yager also introduced two characterizing measures associated with the weight vector $\mathbf{W}$ of an OWA operator: "orness" and "dispersion". The degree of "orness" associated with this operator is defined as:

$$orness(\mathbf{W}) = \frac{1}{n-1} \sum_{i=1}^{n} ((n-i) w_i) \tag{2.20}$$

The "orness" measures how much the aggregation associated with vector $\mathbf{W}$ is like "or"

aggregation operator. The measure of "dispersion" of $\mathbf{W}$ is defined as:

$$dispersion(\mathbf{W}) = -\sum_{i=1}^{n} w_i \ln w_i \tag{2.21}$$

The "dispersion" is a measure of entropy. The method optimizes the variability by maximizing the orness or the dispersion while keeping the dispersion or orness at a fixed level. The weight vectors with different orness and dispersion are evaluated in the cross validation set and the one that gives the highest precision is chosen to be the OWA aggregation operator's weight vector. Li *et al.* used OWA operator fusion in concept detection problems. The features of training set are extracted and different classifiers are trained. Then, by using the outputs of different classifiers as the collection $a_1, a_2, \ldots, a_n$, the weight vectors with different orness and dispersion are evaluated on a cross validation set to find the optimal weight vector. In this way, the OWA aggregation operator is trained and will be used for fusion on test set.

A machine learning fusion model is proposed in [Muneesawang, Guan, and Amin, 2010] for retrieval and classification of movie clips. The audio and visual features are extracted separately and are processed by different similarity functions to obtain the similarity scores. The audio and visual modalities similarity scores are integrated into a two-dimensional vector. SVM is then used to combine the opinions of the different information sources to give a binary decision. The method can be illustrated as in Figure 2.9. It is claimed that it may not be appropriate to directly concatenate audio and visual features into a single representation because of two reasons:

- First, visual feature usually has a physical structure different from audio feature in both dimensions as well as weighting scheme.

- Second, based on previous studies [Massaro, 2001] with respect to human perception, audio and visual processing is likely to be carried out independently in different information sources and combined at a very late stage.

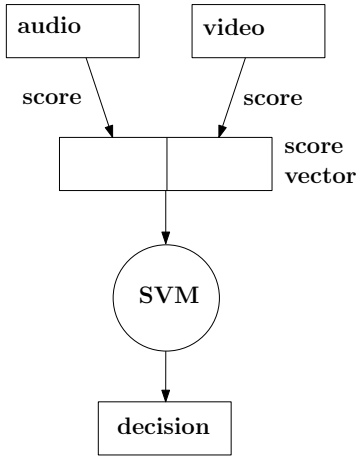


Figure 2.9: Learning fusion model

A multiple feature selection and combination approach for face recognition using Adaboost is proposed in [Contreras, Urunuela, and del Rincon, 2009]. There are two phases of fusion in the approach: feature level fusion and score level fusion. In the feature level fusion, different filters are applied and projected to the PCA/LDA space to obtain new features. A simple substraction of the feature patterns is done, and the resulting features are introduced into the Adaboost algorithm. In the score fusion level, the only difference is that the nearest neighbor matching results in the PCA/LDA space are introduced into the Adaboost algorithm. Thus, the features are considered as a whole in the Adaboost algorithm. Generally, the method transforms different modality features into one whole feature, and then uses the feature in Adaboost. However, the method is not generic for different applications.

For the sake of completeness, there has also been a few classification-based feature fusion methods, for example, [Ross and Govindarajan, 2005; Snoek, Worring, and Smeulders, 2005]. In [Ross and Govindarajan, 2005], the features from biometric sources are normalized using the median normalization scheme. The feature fusion is accomplished by a simple concatenation of the feature sets obtained from multiple information sources. Then, a feature selection relying on an appropriately formulated objective

function is applied to the fused feature to elicit the optimal subset of features from the complete feature set. The match level score is obtained by taking the average of matching score from each source. Euclidean distance and thresholded absolute distance between fused feature vectors are consolidated into one feature level score via average rule. The average of both match level score and feature level score is considered as the final score. However, this method does not allow incompatible feature sets (such as minutiae points of fingerprints and eigen-coefficients of face) to be combined. Snoek *et al.* compared the classification-based feature fusion and decision fusion method for semantic concept detection in [Snoek, Worring, and Smeulders, 2005]. The classification-based feature fusion method is to concatenate unimodal feature vectors to obtain a fused multimedia representation and then rely on supervised learning to classify semantic concepts. On the other hand, the classification-based decision fusion method is to learn semantic concepts directly from unimodal features and then combine the individual scores to yield a final detection score as input for classification using supervised learning. The classification-based feature fusion method requires one learning phase only, but it is difficult to combine features into a common representation. The classification-based decision fusion method requires multiple learning phases (every information source requires a separate supervised learning stage and the combined representation requires an additional learning stage), and may have potential loss of correlation. Based on an experiment of 20 semantic concepts on 184 hours of broadcast video, it is concluded that the classification-based decision fusion method tends to give better performance for most concepts. Moreover, the improvements are more significant when the classification-based feature fusion method performs better. These results suggest that a fusion strategy on a per-concept basis yields an optimal strategy for semantic video analysis [Snoek, Worring, and Smeulders, 2005].

2.2.3 Summary

A summary of all the classification-based fusion works described above is provided in Table 2.2. The classification-based fusion methods can be divided into classification-based feature fusion method and decision fusion method as stated above. In general, the classification-based fusion method does not have the limitations of linear combination functions. Sophisticated fusion models can be learned using classification algorithm. Furthermore, it is very easy to adopt. However, the computation is much higher than linear fusion methods since it usually requires multiple learning phases. It is also difficult to scale. The fusion model needs to be learned again when a new information source is introduced.

2.3 Cascaded Fusion

2.3.1 Overview of methods

The cascaded fusion methods are another category of decision fusion methods. In general, the cascaded fusion methods use the multiple information sources at different stages instead of integrating and processing the multiple information together.

2.3.2 Representative works

According to the study on different video genres in [Li et al., 2000], "news can fall equally into both the informational audio-centric and informational video-centric categories, and can take advantage of a combination of the different indices for effective browsing". Christel *et al.* [Christel, Huang, and Moraveji, 2004] exploited aural and visual cues for interactive video retrieval. Text-based features are the most reliable high-level features applicable in news and documentary video retrieval [Adams et al., 2002]. Thus, the text from ASR (automatic speech recognition) is used first to locate a candidate shot set. Then, less accurate visual features can be used for filtering by users.

Work	**Fusion data**	**Fusion method**	**Correlation**	**Multimedia analysis task**
[Ross and Govindarajan, 2005]	concatenated features	threshold	Not explicitly considered	verify the identity
Adams *et al.* [Adams et al., 2003]	score vector	SVM	Not explicitly considered	concepts detection
Wu *et al.* [Wu et al., 2004]	score vector	SVM	Not explicitly considered	concepts detection
[Zhu, Yeh, and Cheng, 2006]	score vector	SVM	Not explicitly considered	image categorization
[Gao, Lim, and Sun, 2007]	co-occurrence mode features	maximum entropy	Not explicitly considered	concepts detection
[Li et al., 2009]	score vector	Learning weights through cross validation	Not explicitly considered	concepts detection
[Contreras, Urunuela, and del Rincon, 2009]	transformed features	Adaboost	Not explicitly considered	face recognition
[Muneesawang, Guan, and Amin, 2010]	score vector	SVM	Not explicitly considered	retrieval and classification of movie clips

Table 2.2: A list of the representative works in classification-based fusion methods

Tieu and Viola [Tieu and Viola, 2004] used Adaboost for multiple features selection. Multiple features are extracted from one modality data (image). In each iteration of Adaboost, classifiers are trained for each feature and the best feature is selected. The distribution is updated according to the classification results and instances are resampled. The final result is the weighted combination of the features selected from each iteration. A similar multimodal Adaboost algorithm is proposed in [Xue and Ding, 2006] to integrate 3D and 2D information for face localization. Adaboost is used for training a classifier of multimodal features. At each stage of Adaboost, different single-modality features ((R, G, B) Haar-like feature, mean curvature Haar-like feature, *etc.*) are selected and trained as an intermediate classifier, and the distribution is updated accordingly. The final classifier is a combination of these intermediate classifiers. The method can be applied to multimodal data that have same granularity features. It did not discuss how to use the method for multimodal data that have different granularity features, such as, image and audio.

By simply defining the combination method and optimization criterion, genetic algorithm is used to search for the optimized weights of classifiers in [Ruta and Gabrys, 2001]. The optimization criterion is defined as the sum of decisions and true labels over the validation set. The weights can be either 0 (classifier excluded) or 1 (classifier included). Later, a multidimensional genetic algorithm (GA) is proposed in [Gabrys and Ruta, 2006]. The multidimensional selection model (MSM) is designed to represent the selection dimensions of features, classifiers and combiners. Then, mutation and cross-over operations are defined to guarantee that the average performance will not decrease in the subsequent generations. This particular implementation of GA represents a hill-climbing algorithm. Experiments confirm the superiority of the method. The method is designed for Multi-Classifier Systems. Moreover, validation is needed for the optimization fusion after classifier training.

A fusion learning for multimedia streams using a greedy performance driven

algorithm is proposed in [Joshi, Naphade, and Natsev, 2007]. There are a fixed number of boosting iterations. The decision streams are fixed from the beginning. The fusion learning phase is to find the optimal weights for each decision stream. In each iteration, a proportion of data are used for decision fusion learning. In the decision fusion learning phase, the multiple decision streams from the multimodal data are weighted using a greedy hill-climbing algorithm to find the best set of decision stream weights by which the minimum error is obtained. In the end of the boosting, the weights are averaged.

An average precision (AP) based Adaboost fusion for concept detection is proposed in [Cai et al., 2007]. Six visual features are extracted for each keyframe globally or locally. They are Color Histogram (CH), Color Correlogram (CC), Color Moments (CM), Edge Histogram (EH), TextureWavelet (TW) and Texture Cooccurence (TC). Different features are processed independently to build weak classifiers. For each run, the precision is calculated according to the prediction on samples of each weak classifier. Then, the weak classifier with the max precision is chosen as the classifier for this run. The weights of samples, as well as the weight of this classifier, are updated according to this classifier's precision. Sample weights are used to weight the individual average precision of each sample so that the larger weight and former rank the sample has, the more contribution it does to the total average precision of the weak classifier. In the end, the weighted sum of the selected classifiers is used as the strong classifier. Since the non-interpolated average precision is applied as a measure of High Level Feature Extraction at TRECVID, AP-based Adaboost outperforms the standard Adaboost and many other fusion methods in High Level Feature Extraction at TRECVID. In general, it is similar to the method in [Xue and Ding, 2006].

A novel fusion method based on the Combined Adaboost Classifier Ensembles (CACE) algorithm is proposed in [Tan et al., 2009]. The visual features is categorized by different granularities and a pair-wise feature diversity measurement is defined. Then, the simple classifiers based on the feature diversity is constructed. The authors then

use modified AP-based Adaboost to fusion the classifier results.

2.3.3 Summary

A summary of the main cascaded fusion works described above is provided in Table 2.3.

Work	**Stages**	**Fusion stage**	**Correlation**	**Multimedia analysis task**
[Christel, Huang, and Moraveji, 2004]	Filtering using different information sequentially	No	Not considered	interactive video retrieval
[Xue and Ding, 2006]	One modality selected at each stage	Combination of classifiers at all stages	Not considered	face localization
[Joshi, Naphade, and Natsev, 2007]	Greedy hill-climbing algorithm to find the best set of decision stream weights	Average of weights at all stages	Not considered	concepts detection
[Cai et al., 2007]	One modality selected at each stage	weighted sum of the selected classifiers	Not considered	concepts detection
[Tan et al., 2009]	One modality selected at each stage	weighted sum of the selected classifiers	Simple diversity	concepts detection

Table 2.3: A list of the representative works in classification-based fusion methods

2.4 Discussion of decision fusion methods

In sum, there have been several existing decision fusion methods in literature. However, several problems are still open and not well studied. The rule-based fusion methods, especially the linear fusion methods, generally are easy to adopt and scale, and does not need much computation. But how to determine the optimal weights is still an open problem. The classification-based fusion methods are generally more sophisticated, but more computational complex and difficult to scale. The cascaded fusion methods use different information sources sequentially at different stages instead of combining them in one stage. The methods generally overlook the correlation among information sources

and also suffer from high computation cost.

Generally speaking, there are some issues in the previous decision fusion methods:

- First and foremost, the correlation among different information sources is not well utilized to obtain better results. How to differentiate the correlation among information sources and how to utilize the different correlations are seldom considered. A multiple utilization of correlation or a sophisticated correlation model should help to improve the fusion performance.

- In multimedia applications, usually little amount of training data are available at the beginning. The fusion performance may suffer as a result. Furthermore, the multimedia data keep increasing with time. New instances of multimedia data are continuously added. Thus, the evolution of the fusion model is of primary importance because of the nature of multimedia applications. However, few fusion methods have been proposed to cope with the new data well. An evolving fusion method which evolves the multimedia fusion model and improves the performance with new data should be quite useful.

- The information source may have different expertise on different data so that an overall weight for the information source is not suitable. Few data dependent fusion methods have been proposed. A sophisticated data dependent fusion methods that effectively measures the expertise of the output and then adopt it in fusion process should be helpful to improve the performance.

The objective of this dissertation is to develop fusion methods that address these research challenges.

Chapter 3

MultiFusion

Multimodal fusion is useful for many multisensor applications. The utilization of correlations plays an important part and is crucial for multimedia fusion. The appropriate synchronization of the different modalities is still a big research problem [Atrey et al., 2010]. It has also been observed that correlation at the semantic level (decision level) has not been fully explored, although some initial attempts have been reported. Moreover, the correlation of multiple information sources at decision level is usually utilized only once in the whole fusion procedure. In this chapter, a novel multimedia fusion algorithm is proposed which is referred to as "MultiFusion". The approach adopts a boosting structure where the atomic event is considered as the *fusion unit* to process the multimedia data uniformly. The correlation of multimedia information sources is implicitly used to form an overall classifier in each iteration.

3.1 Introduction

In multimedia fusion, one decision can be obtained from each information source. Considering each information source as an expert, the decisions from all the information sources can be combined together to achieve an improved result. In general, classifica-

tion is one of the basic ways to get a decision from each information source.

For each information source in the multimodal data, a classifier can be trained using some supervised learning algorithm. Then, the outputs of the classifiers from each information source will be weighted and combined to develop a final classifier. By assembling multiple classifiers, the ensemble based systems have been shown to produce favorable results compared to the single classifier systems for a broad range of applications and under a variety of scenarios [Polikar, 2006]. There are also several mathematically sound reasons for considering ensemble systems. A set of classifiers with similar training performances may have different generalization performances. Combining the outputs of several classifiers by averaging may reduce the risk of an unfortunate selection of a poorly performing classifier. The averaging may or may not beat the performance of the best classifier in the ensemble, but it definitely reduces the overall risk of making a particularly poor selection. In multimedia fusion, if we have several sets of data obtained from various information sources, where the nature of features is incompatible (heterogeneous features), a single classifier cannot be used to learn the information contained in all of the data. Ensemble based approaches have successfully been used for such applications, such as monologues detection in video shots using both audio and video signals in [Iyengar, Nock, and Neti, 2003a].

There have been several multimedia fusion methods. But how to utilize the different correlations in the multimedia data are seldom considered. A good utilization of correlation among information sources should improve the fusion performance significantly. For the previous multimedia fusion methods, the different information sources are generally fused only at the final combination step. Here, we try to apply fusion and use correlation multiple times by adopting boosting structure and combining the results at each iteration. Based on the Adaboost-like structure, the fusion of these classifiers may lead to an overall performance improvement.

3.1.1 Background

Classification is a machine learning technique for deducing a function from training data. The training data consist of pairs of input objects (typically vectors) and desired outputs. The output of the function can predict a class label of the input object. Machine learning algorithms take a training set, form hypotheses or models, and make predictions about the labels. Because the training set is finite and the future application dataset is uncertain, learning theory usually does not yield absolute guarantees of performance of the algorithms. Instead, probabilistic bounds on the performance of machine learning algorithms are quite common.

One of the popular ensemble based algorithms is Adaboost [Freund and Schapire, 1997]. It boosts a weak learner into a strong learner. Here, the weak learner and strong learner are defined in the PAC (probably approximately correct) model. In this model, the learner tries to identify an unknown concept based on randomly chosen examples of the concept. Examples are chosen according to a fixed but unknown and arbitrary distribution on the space of instances. The learner's task is to find a hypothesis or prediction rule of his own that correctly classifies new instances as positive or negative examples of the concept. With high probability, the hypothesis must be correct for all but an arbitrarily small fraction of the instances [Schapire, 1990].

The notations are defined as follows:

- $\mathcal{C}$ is concept class. $\mathcal{C}$ is decomposed into subclasses $\mathcal{C}_n$ indexed by a parameter n, $\mathcal{C} = \bigcup_{n \geq 1} \mathcal{C}_n$.
- $X = \bigcup_{n \geq 1} X_n$. X_n is the common domain for the concepts in $\mathcal{C}_n$.

A concept class $\mathcal{C}$ is learnable, or strongly learnable, if there exists an algorithm A such that for all $n \geq 1$, for all target concepts $c \in \mathcal{C}_n$, for all distributions D on X_n, and for all $0 < \epsilon, \delta \leq 1$, algorithm A, given parameters n, ϵ, δ, the size s of c, and access to oracle EX (source of examples), runs in time polynomial in ($n, s, \frac{1}{\epsilon}$ and $\frac{1}{delta}$), and

outputs a hypothesis h that with probability at least $1-\delta$ is ϵ-close (the probability of misclassification is no more than ϵ, which is an arbitrarily small fraction) to c under D [Schapire, 1990]. That is, a strong learner generates a classifier that can correctly classify all but an arbitrarily small fraction of the instances. A concept class $\mathcal{C}$ is weakly learnable, if there exists a polynomial p and an algorithm A such that for all $n \geq 1$, for all target concept $c \in \mathcal{C}_n$, for all distributions D on X_n, and for all $0 < \delta \leq 1$, algorithm A, given parameters n, δ, the size s of c, and access to oracle EX, runs in time polynomial in (n, s and $\frac{1}{\delta}$), and outputs a hypothesis h that with probability at least $1-\delta$ is $(1/2 - 1/p(n,s))$-close (the probability of misclassification is no more than $(\frac{1}{2} - \frac{1}{p(n,s)})$, which is less than $\frac{1}{2}$) to c under D [Schapire, 1990]. In other words, a weak learner produces a prediction rule that performs just slightly better than random guessing. It has been proved that the strong and weak learnability are equivalent. A concept class $\mathcal{C}$ is weakly learnable if and only if it is strongly learnable [Schapire, 1990].

Adaboost has many variations. Here, we will mainly discuss Adaboost.M1 [Freund and Schapire, 1996]. The pseudocode of the algorithm is provided in Algorithm 1. Here are some remarks about the symbols:

- $(X_1, Y_1), \ldots, (X_n, Y_n)$ are the training data set. $X_i \in \mathcal{X}$ where $\mathcal{X}$ is the set of data instances, and $Y_i \in \mathcal{Y} = \{\omega_1, \ldots, \omega_C\}$ where $\mathcal{Y}$ is the set of labels, $\omega_1, \ldots, \omega_C$ are the classes of labels.
- **WeakLearn** is a weak learning algorithm defined in the PAC model.
- P_t is the distribution at step t
- H_f is the final hypothesis
- The function $I[Q]$ is Iverson bracket, which is defined as follows:

$$I[Q] = \begin{cases} 1 & \text{if } Q \text{ is true} \\ 0 & \text{otherwise} \end{cases}$$

Input: $(X_1, Y_1), \ldots, (X_n, Y_n)$ where $X_i \in \mathcal{X}$, $Y_i \in \mathcal{Y} = \{\omega_1, \ldots, \omega_C\}$;
Weak Learning algorithm **WeakLearn**;
Initially the distribution $P_1, P_1(i) = \frac{1}{n}$ for $i = 1, \ldots, n$.
Output: H_f
for $t = 1$ *to* T **do**
  Select a training data subset S_t drawn from the distribution P_t;
  Train **WeakLearn** with the training data S_t, and obtain hypothesis $h_t : X \to Y$;
  Calculate the weighted error of h_t: $\varepsilon_t = \sum_{i=1}^{n} P_t(i) I[h_t(X_i) \neq Y_i]$;
  if $\varepsilon_t > 1/2$ **then**
    set $T = t - 1$;
    abort;
  end
  Calculate $\beta_t = \frac{\varepsilon_t}{1-\varepsilon_t}$;
  Update distribution P_t: $P_{t+1}(i) = \frac{P_t(i)}{Z_{t+1}} \times \begin{cases} \beta_t & \text{if } h_t(X_i) = Y_i \\ \frac{1}{\beta_t} & \text{otherwise} \end{cases}$ where
  $Z_t = \sum_{i=1}^{n} P_t(i)$ is a normalization factor;
end
$H_f(X) = \arg\max_{Y \in \mathcal{Y}} \sum_{t=1}^{T} (\log \frac{1}{\beta_t}) I[h_t(X) = Y]$

Algorithm 1: Adaboost

Adaboost takes a weak learning algorithm and a sequence of instances initially. Then, Adaboost generates a set of hypotheses by training a weak classifier using instances drawn from an iteratively updated distribution of the training data. The distribution update ensures that instances misclassified by the previous classifier are more likely to be included in the training set of the next classifier. Hence, consecutive classifiers' training data are geared towards increasingly hard-to-classify instances. In the end, all the hypotheses generated at each step are combined through weighted majority voting of the classes predicted by the individual hypothesis. Thus, Adaboost is a way to boost a weak learning algorithm to a strong learning algorithm.

3.1.2 Related Work

Polikar *et al.* introduced a supervised incremental learning algorithm Learn++ [Polikar et al., 2001]. Here, the ability of a classifier to learn with subsequently acquired datasets and acquire the newly introduced knowledge without forgetting the previously learnt information is usually referred to as incremental learning. The algorithm is based on generating a number of hypotheses using different distributions of the training data and combining these hypotheses using a weighted majority voting. Later on, Parikh *et al.* applied Learn++ to the multimodal fusion problem in the method of combining classifiers [Parikh et al., 2004]. Incremental learning and multimodal fusion are conceptually similar: in incremental learning, the algorithm will learn from multiple datasets and the datasets may introduce new classes. In multimodal fusion, the algorithm will combine multiple modalities datasets and the datasets may contain different features. Learn++ is used to train the classifier for information fusion. The classifier is trained from each modality data in the way similar to Adaboost. Then, the different classifiers from different modality datasets are combined together using weighted majority voting. Here, without newly introduced data, we adopt Adaboost to train classifiers for different modalities and combine together using weighted majority voting. The fusion method is illustrated in Figure 2.6. For clarity, it is referred to as *Learn++ fusion* in the rest of this chapter.

Learn++ boosts the classifiers from different datasets using Adaboost and the decisions from different datasets are combined using weighted majority voting. The decisions have the same representation. Moreover, the algorithm can be scaled in terms of the modalities used in the fusion process. However, there are some disadvantages for Learn++ in multimedia fusion. The different classifiers are trained to obtain the local decisions which are combined in the end. It is likely that the final decision depends more on the accuracy of each modality instead of the combination of multiple modali-

ties. The correlation of the multimodal data is used only at the final combination step. Thus, one of the disadvantages lies in its failure to fully utilize the correlation among modalities. If we can shift the combination of multiple modalities to each step of boosting, the correlations among modalities can be made better use of, thus obtaining better combination results at each step. Moreover, by doing this in the boosting structure, we are actually improving the overall utilization of multimodal data instead of individual performance of each modality. It can therefore improve the fusion performance.

The Adaboost algorithm has already been used in many unimodal applications, such as [Guo and Zhang, 2001], [Zhang, Li, and Zhang, 2002], [Pickering, Ruger, and Sinclair, 2002], [Tieu and Viola, 2004]. Guo *et al.* [Guo and Zhang, 2001] directly applied Adaboost with face features to boost the face recognition results. Some other methods [Amores et al., 2004], [Zhang, Li, and Zhang, 2002], [Tieu and Viola, 2004] used Adaboost for multiple features selection. The method in [Zhang, Li, and Zhang, 2002] used Adaboost for each feature, while the method in [Tieu and Viola, 2004] compared the different features and selected one best feature at each step of Adaboost. In [Tieu and Viola, 2004], multiple features are extracted from one modality data (image). In each iteration of Adaboost, classifiers are trained for each feature and the best feature is selected. The distribution is updated and instances are resampled. The final result is the weighted combination of the features selected from each iteration. Barbu *et al.* [Barbu, Iqbal, and Peng, 2005] used similar methods to [Tieu and Viola, 2004]. Guo *et al.* proposed a method of SAR image target recognition based on Adaboost algorithm in [Guo et al., 2008]. They adopt the Adaboost algorithm for feature selection. Initially, there is a multi-dimension feature generated from the image. For each dimension, an optimal threshold is selected such that the minimum number of instances are misclassified. Each thresholded single feature is viewed as a linear binary classifier. For the Adaboost feature selection procedure, the classifier is a weighted combination of all the thresholded single features according to the probability. The final classifier is the combi-

nation of each intermediate classifier. In this way, the weight of each feature dimension is determined and the effective features are selected according to the weights.

There are also a few similar works in the multimodal analysis area. Maghooli *et al.* [Maghooli and Moin, 2004] proposed a novel approach in the field of multimodal biometrics based on Adaboost. The multimodal features are fused together, and a weak learner (only one neuron neural network) is used in the Adaboost. A multimodal Adaboost algorithm similar to [Tieu and Viola, 2004] is proposed in [Xue and Ding, 2006] to integrate 3D and 2D information. Adaboost is used for training a classifier of multimodal features. At each stage of Adaboost, different single-modality features ((R, G, B) Haar-like feature, mean curvature Haar-like feature, *etc.*) are selected and trained as an intermediate classifier, and the distribution updated accordingly. The final classifier is a combination of these different classifiers. The method can be applied to multimodal data that have same granularity (or compatible) features. It did not discuss how to use the method for multimodal data that have different granularity features, such as, image and audio. However, by using the fusion unit we introduced in this chapter, the method can be used for fusion of various heterogeneous multimodal data. We refer to this method as the *Selection fusion* method. A fusion learning for multimedia streams using a greedy performance driven algorithm is proposed in [Joshi, Naphade, and Natsev, 2007]. There are a fixed number of boosting iterations. The decision streams are fixed from the beginning. The fusion learning phase is to find the optimal weights for each decision stream. In each iteration, a proportion of data are used for decision fusion learning. In the decision fusion learning phase, the multiple decision streams from the multimodal data are weighted using a greedy hill-climbing algorithm to find the best set of decision stream weights by which the minimum error is obtained. In the end of the boosting, the weights are averaged. A multiple feature selection and combination approach for face recognition using Adaboost is proposed in [Contreras, Urunuela, and del Rincon, 2009]. There are two phases of fusion in the approach: feature

level fusion and score level fusion. In the feature level fusion, different filters are applied and projected to the PCA/LDA space to obtain new features. A simple substraction of the feature patterns is done, and the resulting features are introduced into the Adaboost algorithm. In the score fusion level, the only difference is that the nearest neighbor matching results in the PCA/LDA space are introduced into the Adaboost algorithm. Thus, the features are considered as a whole in the Adaboost algorithm. Generally, the method transforms different modality features into one whole feature, and then uses the feature in Adaboost. However, the method is not generic for different applications.

To make use of the advantages of multimodal data and the correlations among the information sources, a novel multimedia fusion approach is proposed here, which is referred to as *MultiFusion*. The MultiFusion algorithm utilizes multiple fusion in a way similar to Adaboost. It fuses the multimodal data using weighted majority voting in each step, thus the correlations among multimodal data are implicitly utilized in every individual step. In this way, it improves the overall fusion performance instead of individual modality performance. In the following, we first state the proposed algorithm in details in Section 3.2. Then, both the simulation and real application results are shown in Section 3.3 to demonstrate the improvements. Finally, the conclusion is given in Section 3.4.

3.2 Proposed Algorithm

Our proposed multimodal fusion algorithm adopts the decision level fusion strategy [Wang and Kankanhalli, 2010a]. It represents the decisions at the semantic level in the same form. The scalability can also be achieved depending upon the information sources used. Moreover, it overcomes the disadvantages of single time fusion (such as Learn++ fusion) by introducing the notion of a "fusion unit" and fusing using weighted majority voting at each step to implicitly utilize the correlation among information sources. In the

single time fusion approach, the multimodal data are fused only at the final combination step. Thus, it fails to fully utilize the correlation among information sources. In contrast, each fusion unit contains multiple information sources and can be dealt with uniformly. It can work with heterogeneous features since each information source of fusion unit can be trained using an appropriate method. Moreover, the correlation among information sources is better utilized by fusing at each step based on the fusion unit. Unlike the unimodal Adaboost algorithm, our proposed fusion method is for multimodal fusion in one integrated framework. By introducing atomic events as the fusion unit, the correlations among the various information sources are utilized by fusing the multimodal data at each step. Moreover, by adopting the Adaboost-like structure, the fusion of these classifiers leads to an overall performance improvement.

3.2.1 Data Representation

First of all, we need to specify the representation of the input data to the multimodal fusion algorithm. Multimodal data are heterogeneous. Thus, the management of the multimodal data is also related to multimodal fusion. There are generally two kinds of approaches of managing multimodal data: media-centric approach and event-centric approach. The media-centric approach manages the multimodal data according to the data type. Providing an alternative perspective, the event-centric approach manages the multimodal data in the form of events. Here, the appropriate method of managing multimodal data is the event-centric method. The multimodal data from each information source will be synchronized in the preprocessing stage. Then, the synchronized multimodal data will be segmented into a sequence of segments, say, events. The concepts are defined in [Atrey, Kankanhalli, and Jain, 2006]. 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). Atomic event is an event in which exactly one

object having one or more attributes is involved in exactly one activity. Compound event is the composition of two or more different atomic events. In multimodal data, each atomic event is associated with parts of the data of each information source. For example, the compound event "A person ran through the corridor, and then entered the meeting room" consists of two atomic events "a person ran through the corridor" followed by "person entered the meeting room". The detection of the atomic events requires multimodal data. For example, a "running" event can be detected based on both video and audio streams. In image retrieval, the atomic event is an image object which contains a single image for image modality and comments about the image for text modality. The image can be retrieved based on both image and text content.

For each application, suppose we have a multimodal dataset $\mathcal{M}$ which contains N information sources, *i.e.*, $M_k, k = 1, 2, \ldots, N$. The information sources can be video, audio, image, text, *etc.* The multimodal data are heterogeneous and organized on a timeline. Thus, the data set will be segmented into unit segments, *i.e.*, atomic events, and each atomic event is one dataset instance which consists of a segment of data of each modality. The atomic events are used to define the basic fusion granularity, *i.e.*, the **fusion unit**. The multimodal data express a sequence of events. Each event is associated with parts of each modality data. The granularity is different for each modality and is based on the nature of different modality. For example, for the concept detection application in Flickr images, the multimodal data are the images and the corresponding comments over the site. Then, the atomic event is a single image for image modality and comments about the image for text modality. Similarly, for the Youtube video search, the multimodal data are the videos and text descriptions. Then, the atomic event is a single video clip for video modality and corresponding description words for text modality.

The multimodal dataset is illustrated in Figure 3.1. For a multimodal dataset $\mathcal{M}$, it contains N modalities $M_1, \ldots, M_N$. For example, in a multimodal surveillance

data set, there may be different information sources, such as, infrared camera video, audio, and video. The multimodal dataset can be segmented into several atomic events. The atomic event is taken as the basic fusion unit. In a multimodal surveillance dataset, the fusion unit may be a shot.

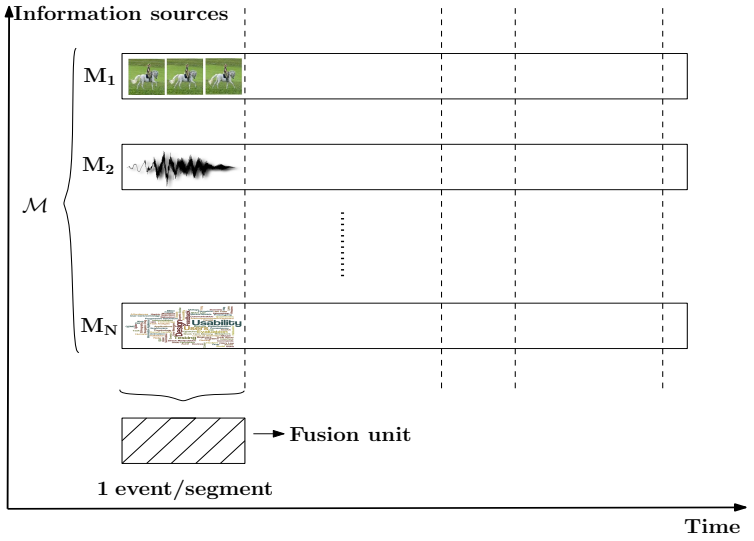


Figure 3.1: The illustration of the data representation

3.2.2 Fusion Phases

The proposed multimodal fusion algorithm consists of two phases: training and testing. The multimodal data will first be segmented into fusion units. In the training phase, the input will be training datasets consisting fusion units $\mathcal{X}$ with corresponding labels $\mathcal{Y}$. After the training process, the output will be a final classifier $H_{final} : \mathcal{X} \rightarrow \mathcal{Y}$. Then, in the testing phase, the input is the set of multimodal fusion units data, and the output is the label for each input fusion unit by using the final classifier. The two phases of the proposed multimodal fusion algorithm are illustrated in Figure 3.2.

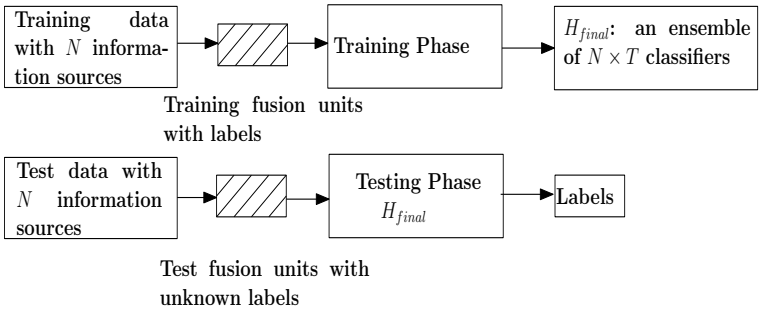


Figure 3.2: The illustration of the fusion phases

3.2.3 Fusion Algorithm Description

The *MultiFusion* algorithm is illustrated in Figure 3.3, and the details of the algorithm are described in Algorithm 2.

- Inputs to our method are a sequence of labeled fusion units (training data) from each of the dataset of different information sources
- **metaFusion** denotes a classifier learning algorithm. It can be SVM, KNN, *etc.*. **metaFusion** is similar to *WeakLearn* in Adaboost which has already been defined in the PAC model.
- T is an integer which specifies the number of classifiers (iterations) to be generated by **metaFusion** for each dataset.

In the proposed method, the different labeled instances of the same position will correspond to the same segment in the multimodal dataset, which means the multimodal data are synchronized. The multimodal data will be segmented into uniform units. Each instance of training fusion units data contains different modalities of the multimodal data and **metaFusion** will use different modality data in $\mathcal{X}$ to train classifiers for different modality $M_k, k = 1, 2, \ldots, N$.

Our proposed algorithm iteratively updates the distribution of training data by assigning appropriate weights to each fusion unit such that the classifier of **metaFusion**,

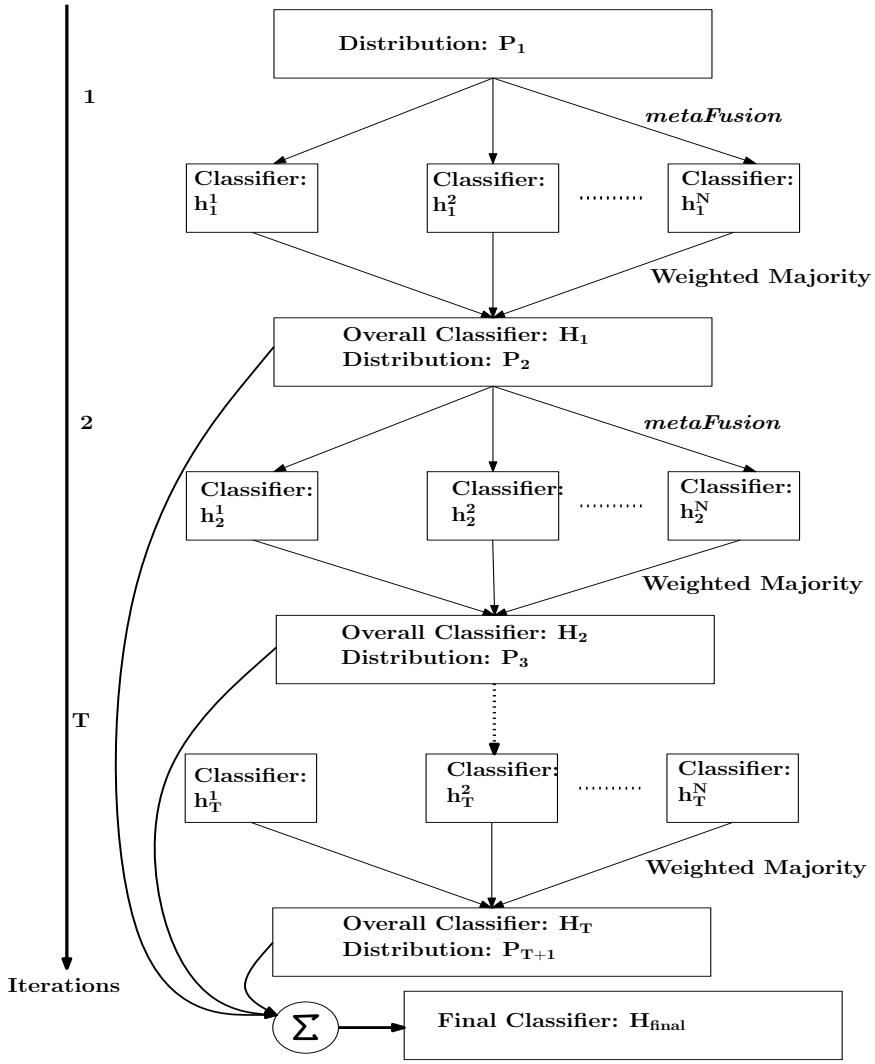


Figure 3.3: The illustration of the proposed fusion method. There are T iterations. At each iteration t, N different information sources of fusion units are trained separately to obtain multiple classifiers $h_t^1, \ldots, h_t^N$. The classifiers are combined using weighted majority to get an overall classifier H_t for the fusion units. The distribution is updated to P_{t+1} and used for re-sampling fusion units in the next iteration. In this way, the re-sampling at each iteration is aimed to boost the overall classifier for the fusion units. Finally, the classifiers at each iteration $H_1, \ldots, H_T$ are combined to develop the final classifier H_{final}.

Input: The datasets drawn from information sources $M_k, k = 1, 2, \ldots, N$;
Sequence of n fusion units data $S = \{(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n)\}$,
where $X_i \in \mathcal{X}$, $Y_i \in \mathcal{Y} = \{\omega_1, \ldots, \omega_C\}$;
Basic fusion classifier learning algorithm *metaFusion*;
Integer T, specifying the number of iterations;
Initialize the distribution $P_1(i) = \frac{1}{n}, \forall i$
Output: The final classifier: $H_{final} : \mathcal{X} \to \mathcal{Y}$
for $t = 1$ *to* T **do**
  Select a training fusion unit subset S_t drawn from the distribution P_t;
  for $k = 1$ *to* N **do**
    Select the corresponding data S_t^k in information source M_k from fusion units subset S_t;
    Call *metaFusion*, providing it with the training data S_t^k, and obtain a classifier $h_t^k : \mathcal{X} \to \mathcal{Y}$;
    Calculate the weighted error of h_t^k: $\varepsilon_t^k = \sum_{i=1}^{n} P_t(i) I[h_t^k(X_i) \neq Y_i]$;
    if $\varepsilon_t^k > 1/2$ **then**
      set $T = t - 1$;
      discard h_t^k and abort;
    end
    Calculate scaled error $\beta_t^k = \frac{\varepsilon_t^k}{1-\varepsilon_t^k}$;
  end
  Call weighted majority, obtain the overall classifier $H_t : \mathcal{X} \to \mathcal{Y}$:
  $H_t(X) = \arg\max_{Y \in \mathcal{Y}} \sum_{k=1}^{N} (\log \frac{1}{\beta_t^k}) I[h_t^k(X) = Y]$;
  Compute the overall error $E_t = \sum_{i=1}^{n} P_t(i) I[H_t(X_i) \neq Y_i]$;
  Calculate scaled error $B_t = \frac{E_t}{1-E_t}$;
  Update the distribution P_t:

$$P_{t+1}(i) = \frac{P_t(i)}{Z_{t+1}} \times \begin{cases} B_t, & \text{if } H_t(X_i) = Y_i \\ \frac{1}{B_t}, & \text{otherwise} \end{cases}$$

  where $Z_t = \sum_i P_t(i)$ is a normalization constant such that P_{t+1} will be a distribution;
end
Call Weighted majority and the final classifier:
$H_{final}(X) = \arg\max_{Y \in \mathcal{Y}} \sum_{t=1}^{T} (\log \frac{1}{B_t}) I[H_t(X) = Y]$

Algorithm 2: MultiFusion – The Proposed Multimedia Fusion Method

which is trained with a subset training data drawn from this distribution, is forced to focus on increasingly harder fusion units for the overall classifiers H_t.

At iteration t (T iterations in total), our algorithm provides **metaFusion** with a subset training data drawn according to distribution P_t from the original training data $S = \{(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n)\}$, where X_i are training fusion units data and Y_i are the correct labels for fusion units $i = 1, \ldots, n$. **metaFusion** then computes a classifier $h_t^k : \mathcal{X} \rightarrow \mathcal{Y}$, which should correctly classify a fraction of the training set with respect to P_t. That is, **metaFusion**'s goal is to find a classifier h_t^k, which minimizes the training error:

$$\varepsilon_t^k = \sum_{i=1}^{n} P_t(i) I[h_t^k(X_i) \neq Y_i]$$

and scaled error is calculated as

$$\beta_t^k = \frac{\varepsilon_t^k}{1 - \varepsilon_t^k}$$

Similar to Adaboost, it requires that $\varepsilon_t^k < 1/2$ for each h_t^k, that is, each classifier must obtain a minimum performance of 50%. If $\varepsilon_t^k > 1/2$, the **metaFusion** will abort. For each modality dataset, h_t^k is obtained. Then, the overall classifier is obtained using weighted majority voting

$$H_t(X) = \arg\max_{Y \in \mathcal{Y}} \sum_{k=1}^{N} (\log \frac{1}{\beta_t^k}) I[h_t^k(X) = Y]$$

and the error is calculated as:

$$E_t = \sum_{i=1}^{N} P_t(i) I[H_t(X_i) \neq Y_i]$$

The initial distribution P_t is uniform over S, that is, $P_t(i) = \frac{1}{n}$ (the probability for fusion unit X_i). All fusion units in S are therefore equally likely to be drawn as the subset training data. The distribution is updated by

$$P_{t+1}(i) = \frac{P_t(i)}{Z_{t+1}} \times \begin{cases} B_t, & \text{if } H_t(X_i) = Y_i \\ \frac{1}{B_t}, & \text{otherwise} \end{cases}$$

where

$$B_t = \frac{E_t}{1 - E_t}$$

and $Z_t = \sum_i P_t(i)$ is a normalization constant such that P_{t+1} will be a distribution.

Essentially, easy fusion units that are correctly classified by H_t get a lower probability, and hard fusion units that are misclassified have a higher probability of being selected for the next training data subset. Thus, our algorithm focuses on the fusion units that seem to be hardest for **metaFusion** to train the overall classifier.

At the end of T iterations, the algorithm combines the intermediate classifiers $H_1, \ldots, H_T$ (the fusion classifier at each iteration) into a single final classifier H_{final} using weighted majority voting.

$$H_{final}(X) = \arg\max_{Y \in \mathcal{Y}} \sum_{t=1}^{T} (\log \frac{1}{B_t}) I[H_t(X) = Y]$$

There are n fusion units of N information sources. Suppose the complexity of **metaFusion** is α. If we adopt SVM, the complexity of **metaFusion** is $\alpha = O(n^3)$. The complexity of each iteration is then $O(Nn^3)$. There are T iterations in our fusion method. Then, the total complexity is $O(TNn^3)$. It is the same as Learn++ fusion method. Usually, $T, N \ll n$.

3.2.4 Remarks

We make the following observations about our proposed MultiFusion algorithm.

- In the proposed fusion algorithm, **metaFusion** denotes a classifier training algorithm, such as SVM. For the N information sources and T iteration steps, the **metaFusion** classifier can be the same or different for different information sources. Thus, it can be suitably adapted.

- A classifier is a mapping from a feature space $\mathcal{X}$ to a discrete set of labels $\mathcal{Y}$. Thus, for each modality dataset M_k, at each step t, though the feature space and discrete

set of labels are the same, the mapping will be different. Thus, the classifiers are different at each step even for the same modality dataset. As a result, for the N information sources and T iteration steps, the total number of different classifiers are $N \times T$.

- At each step t, all the N classifiers for all the information sources will be trained and then fused using *weighted majority voting*. In this way, a fused classifier H_t for N information sources at step t is obtained and stored. After all the T iteration steps, T fused classifiers $H_t, t = 1, 2, \ldots, T$ will be obtained. The final classifier H_{final} is a combination of these fused classifiers using *weighted majority voting*. After the training phase, for each data instance $X \in \mathcal{X}$, the final label Y is the *weighted majority voting* of $H_t(X), t = 1, 2, \ldots, T$ as for the H_{final}.

- The iteration steps T is user defined. Boosting forever can overfit the data and therefore in order to achieve consistency, it is necessary to stop the boosting procedure early after a very small number of iterations, such as 10 or 100 [Zhang and Yu, 2005].

3.2.5 Comparison

It can be seen that there are several boosting methods for both unimodal and multimodal data. The unimodal methods are not be suitable for multimodal data due to the heterogeneity of the data and features. Some of the previous multimodal algorithms using Adaboost fuse the features and use the combined features as the input for Adaboost. It may be suitable only for some features since not all the features are homogeneous and compatible to be combined together. It may be difficult to scale them for too many features due to the high dimensionality of the combined feature. The related works are summarized and compared in Table 3.1.

As shown in the table, our proposed MultiFusion algorithm has some advantages.

Methods	**Modality**	**Fusion level**	**Intermediate samples**	**Intermediate output**	**Modalities used**
[Tieu and Viola, 2004]	Unimodal	Decision	Resampled instances	One feature classifier	One
[Maghooli and Moin, 2004]	Multimodal	Feature	Fused multimodal features	Classifier for fused multimodal feature	All
Learn++ Fusion [Parikh et al., 2004]	Multimodal	Decision	Resampled data for each modality	Classifier for each modality	All
Selection Fusion [Xue and Ding, 2006]	Multimodal	Decision	One feature of resampled instances	One feature classifier	All
[Joshi, Naphade, and Natsev, 2007]	Multimodal	Decision	Partial bagging data & Fixed classifiers	Weights of classifiers	Part
[Contreras, Urunuela, and del Rincon, 2009]	Multimodal	Feature	Resampled data for whole integrated feature	Classifier for integrated feature	All
MultiFusion method	Multimodal	Decision	Resampled multimodal instances	Weighted multimodal classifiers	All

Table 3.1: Comparison of proposed algorithm with representative related existing fusion algorithms

First of all, it works for multimodal data. We propose to use atomic event as the fusion unit for multimodal data. This allows for each fusion unit to contain multiple information sources and can be dealt with uniformly. It can work with heterogeneous features since each information source of fusion unit can be trained using an appropriate method. Second, it is a decision fusion method. The algorithm is thus easily scalable since only decisions from all information sources are combined. Third, it adopts the Adaboost structure and makes good use of the correlations among the information sources. By a simple weighted combination of the multimodal data, the correlation of multimodal data is used to develop one overall fusion decision at each iteration. More complicated forms of multimodal data combination based on the data correlation can be applied without much change. Then by updating the distribution and re-sampling the data, the overall fusion performance should be improved because of the boosting structure. Moreover, the proposed algorithm utilizes information from all of the information sources.

3.3 Experiments and Results

In order to show the effectiveness, our proposed fusion method has been tested on both simulated data and a real application task.

3.3.1 Simulation

To show the effectiveness of the MultiFusion method, we simulated different cases of multiple modalities and compared with different fusion methods. The notations are as follows:

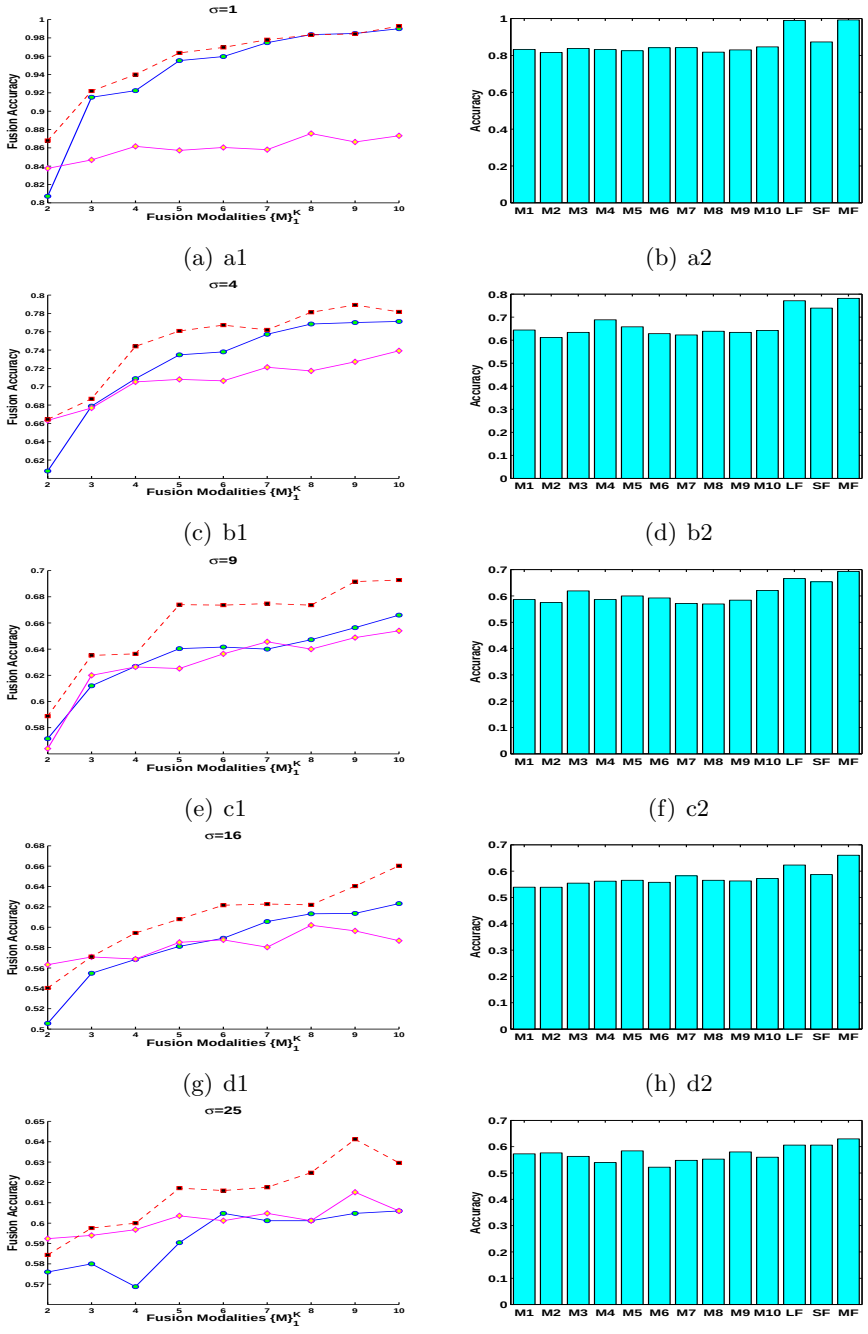
- Let $M_1, M_2, \ldots, M_N$ represent the set of N information sources.
- Let $\{M\}_1^K, K = 1, 2, \ldots, N$ represent the fusion results of modalities 1 to K.
- Let $\mathcal{N}(\mu, \sigma)$ represent a Gaussian distribution with mean μ and variance σ.

We compared our fusion method with the widely-used and related method: one is Learn++ fusion [Parikh et al., 2004], and the other one is Selection fusion [Xue and Ding, 2006], which selects one best single-modality feature to train an intermediate classifier at each stage, and combines these different classifiers to obtain the final results. We performed our simulation on the fusion of N different information sources. We simulated N modalities $M_1, M_2, \ldots, M_N$ and then fused them one by one. For each modality, we assume the data for each class obey the normal distribution [Aly and Hiemstra, 2009]. Thus, the data samples are generated using the following equation:

$$X = \mu + \epsilon \tag{3.1}$$

Here, μ is the mean value for the corresponding class. ϵ is the noise which follows normal distribution $\mathcal{N}(0, \sigma)$ (mean 0 and variance σ). For each modality, there are binary classes. Each class follows the above distribution with different mean value with distance 2. Then, a different σ value will generate data of a different noise characteristic. Smaller variance σ value represents clearer data with less confusion, and large σ generates quite noisy data. The number of instances is $SP = SN = 50$. To reduce the effects of randomness in the results, we repeated every simulation run $L = 50$ times. The results of each simulation run are actually obtained as an average over 50 times. In order to test the results on different noise level, we have five σ values, $\sigma = 1, 4, 9, 16, 25$. Based on these generated data, we simulated the fusion of N modalities. Here, N ranges from 1 to 10. The actual simulation process is described in Algorithm 3.

The simulation results are shown in Figure 3.4. It should be noted that the fusion performance may be degraded when a new noisy information source is introduced. However, the fusion results still generally outperform each information source. As can be seen, the MultiFusion generally outperforms the Learn++ fusion method by 3-9%. When there is less noise in the data and more information sources available, both methods obtain good fusion results and the improvement does not seem to be much. For



(a) a1 (b) a2

(c) b1 (d) b2

(e) c1 (f) c2

(g) d1 (h) d2

(i) e1 (j) e2

Figure 3.4: Simulation results. (a1-e1) shows the simulation fusion accuracy for MultiFusion and weighted majority voting with 1-10 information sources and data distribution variance σ ranging from 1 to 25. In each figure, the red dash line with black square represents the MultiFusion accuracy, the blue line with green circle represents the Learn++ fusion accuracy, and the magenta line with yellow diamond represents the Selection fusion accuracy. (a2-e2) shows the simulation accuracy for 10 modalities fusion. $M_1 - M_{10}$ represents the accuracy for each information source, LF represents the Learn++ fusion, SF represents the Selection fusion, and MF represents the MultiFusion result.

Input: σ for n modalities
Output: Fusion results for $\{M\}_1^K, K = 1, \ldots, N$
foreach *Repetition* $l \in [1..L]$ **do**
  foreach *Modality* M_i **do**
    `// Data generation`
    Generate SN negative samples and SP positive samples;
    Select 50% negative and positive samples as training dataset;
    Use remaining negative and positive samples as test dataset;
  end
  `// Modality fusion`
  foreach *Combination of modalities* $\{M\}_1^K$ **do**
    Train a MultiFusion model for modalities $\{M\}_1^K$ using the training data;
    Test fusion performance and report the results;
  end
end
Report the average achieved performance over L repetitions;

Algorithm 3: The Simulation Procedure

example, the MultiFusion method only outperforms the Learn++ fusion method by no more than 1% with noise $\sigma = 1$ and 10 information sources as shown in Figure 3.4(a). But as the noise in the data increases, the improvement becomes obvious. For example, the MultiFusion method outperforms the Learn++ fusion method by 6% with $\sigma = 16$ using all the information sources. On the other hand, the Selection fusion method generally gets comparable results when the modalities are few (generally less than 4), while the results do not improve much and are generally worse than Learn++ fusion and MultiFusion as the number of information sources increases. Our proposed MultiFusion method generally outperforms the Selection fusion method by 2-11%.

The significance test has also been done for the proposed MultiFusion method. The final fusion results of fusing 10 information sources for different scenarios are collected. By combining all the fusion accuracies of the 50 repetitions, paired *t-test*s have been done between MultiFusion method and Learn++ fusion or selection fusion method. It shows that MultiFusion method has passed the *t-test* with Learn++ fusion at the 5% significance level. The $p - value$ is 0.002. As well, MultiFusion method passed the

(a) human in the region(distant view)

(b) human in the region but occluded

(c) human outside the region

(d) human in the region(close view)

Figure 3.5: Human detection application

t-test with Selection fusion at the 5% significance level.

3.3.2 Human detection

To test the proposed MultiFusion method for a real application, we evaluated it for a human detection task. The data are obtained from AVSS challenges set (`http://www.elec.qmul.ac.uk/staffinfo/andrea/avss2007_d.html`). We chose a dataset containing both video and two-channel audio for multimodal fusion. Preview images can be seen in Figure 3.5. The video is 8-bit color AVI format with 360×288 pixels resolution. Video sampling rate is 25 Hz and audio sampling rate is 44.1 kHz. Sensor

details are as follows:

- The camera is placed in the center of a bar that supports two microphones
- Distance between the microphones: 95 cm
- Microphones: Beyerdynamic MCE 530 condenser microphones
- Camera: KOBI KF-31CD analog CCD surveillance camera

The task is to detect whether there is a human in the rectangle region of the frame, as shown in Figure 3.5.

The data are first segmented into frames and corresponding audio samples as the atomic event. There are 1,077 frames. We randomly select 100 frames as training samples, and the remaining frames are treated as the test set. The features used for human detection are as follows:

- Audio: the ratio and differences of audio energy of two different channels. For each frame, the audio energy for left and right channel E_L and E_R can be easily calculated using audio samples. Then, the ratio of audio energy is obtained as $R = \frac{E_L}{E_R}$, and the difference of audio energy is calculated as $D = E_L - E_R$.
- Visual: the frame difference with the background. The background frame of the scene B is chosen. Then, the gray scale frame difference between any frame F and background B is calculated as $fd = F - B$.

In this experiment, we compared our proposed method with Learn++ fusion and Selection fusion methods. Both methods utilize the Adaboost structure.

Figure 3.6 shows the results. There are only two information sources in the task. Moreover, both features are quite noisy and do not perform well (the audio modality classification accuracy is only 48%, and visual modality classification accuracy is 51%). Sometimes they may conflict with each other. However, the MultiFusion approach still

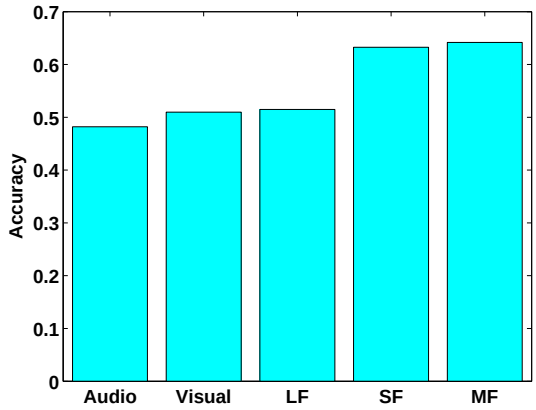


Figure 3.6: The results of human detection on AVSS dataset. *Audio* denotes the audio modality accuracy. *Visual* denotes the visual modality accuracy. *LF* represents the Learn++ fusion results. *SF* represents the Selection fusion results. *MF* represents the MultiFusion results.

performs a little better than the Learn++ fusion and Selection fusion approach. The Learn++ fusion approach obtains a result with about 52% accuracy, the Selection fusion obtains a result with about 63% accuracy, while the MultiFusion approach achieves about 64% accuracy. The proposed MultiFusion method outperforms Learn++ fusion and Selection fusion by 12% and 1% respectively.

3.3.3 Discussion

The resampling procedure can help to improve the performance in Adaboost-like structure. In our proposed MultiFusion method, instead of resampling the instances according to the performance of individual information source, the data are resampled based on the performance of the fusion classifier. Thus, the procedure focuses more on the instances that are difficult for fusion classifier instead of individual information source classifier. For example, in the human detection task, the audio and visual information is used. The difficult instances for single source classifier may not be difficult for the fusion classifier. The human utters sound from time to time. Thus, the case when the human

does not utter sound will be difficult for audio classifier, but it may not be difficult for the fusion classifier since the human can be visually seen sometimes. Similarly, the human can be occluded sometimes. Thus, it is difficult for visual classifier, but it may not be difficult for fusion classifier since the human can be detected by audio. For Learn++, the procedure will focus more on the difficult cases for individual information source. But for MultiFusion, the resampling procedure focuses on the difficult cases for fusion classifier, which is the real difficult cases for fusion. The performance of MultiFusion is thus better than Learn++ fusion. For Selection Fusion, only one information source is used in each iteration, and the others are discarded. Thus, the performance of Selection Fusion is also not as good as that of MultiFusion.

3.4 Conclusions

In this chapter, a novel multimodal fusion algorithm is proposed. The MultiFusion approach is used for multimodal data by segmenting multimodal data into atomic events and using the atomic event as fusion unit. It adopts a boosting structure, where in each iteration, the correlation of multiple information sources is implicitly used by combining different modality classifiers using weighted majority voting to form an overall classifier. The correlations are implicitly used multiple times. Moreover, by adopting the Adaboost-like structure, the overall performance is improved. In this way, the multimodal fusion can be applied to multimodal data in different applications to utilize complementary yet correlated information and improve the performance. Both the simulation experiment and the real application task show the effectiveness of the algorithm. The measurement of multimodal correlations still needs to be studied in detail. More sophisticated fusion techniques to make better use of correlations for improved fusion results will be investigated in the future. Chapter 4 can be one of the possible ways to properly measure and utilize correlation to improve multimedia fusion performance.

Chapter 4

Portfolio Fusion

Various ways of combining the evidences from different information sources have been proposed as discussed in literature review. The correlation among information sources should be useful, but is seldom explicitly considered. Chapter 3 has shown that multiple utilization of correlation can help to improve the performance. A sophisticated modeling of correlation should also help to find the optimal weights in linear fusion.

Linear Opinion Pool is one of the simplest and most widely used methods for combining information from a multiplicity of information sources. As discussed in Chapter 2, the linear fusion method is computationally less expensive compared to other methods. It is also easily scalable. Several methods based on linear fusion have been proposed. Max, Min, Average [Ngo et al., 2007] and Adaboost [Freund and Schapire, 1997] are such commonly used methods. Max, Min, Average [Ngo et al., 2007] fusion methods do not take the difference between modality performances into account. Adaboost fusion obtains the final decision by assigning appropriate weights to each information source. The weights are related to the accuracy-error rate of each information source.

Most of the methods consider the fusion as an information aggregation task. They aim to maximize the aggregated information by assigning proper weights to individual information channels [Li et al., 2009]. However, how to find the appropriate

weights (or confidence level) for different information sources is still an open research issue [Atrey et al., 2010]. Furthermore, the existing methods generally aim to maximize the accuracy, but perfect classification models generally cannot be learned due to noise and the training/test distribution gap. Moreover, in a multimedia data-understanding task, we often assert similarity between data based on our beliefs which does not come from classical probability experiments [Wu et al., 2004]. Thus, the decision cannot be estimated with absolute certainty using the classification models. Thus, the risk (uncertainty) is an intrinsic feature prediction using classification models. The uncertainty is the lack of complete certainty, that is, the existence of more than one possibility. There are many sources of uncertainty such as ambiguity, noise, and deviations between the scoring function and the true probability of relevance. Taking the real accuracy as a type of "an investment return" of our classification models, we should maximize the return as a desirable thing and minimize the variance of the return as an undesirable thing. To the best of our knowledge, minimizing the effect of uncertainties has never been explicitly considered in multimedia fusion methods. To solve this problem, the portfolio theory is introduced.

4.1 Portfolio Theory

Portfolio theory was introduced by Markowitz in [Markowitz, 1952], for which he won the Nobel prize in economics. In his work "Portfolio Theory Selection", he recommended the use of expected return-variance of return rule, *"... both as a hypothesis to explain well-established investment behavior and as a maxim to guide one's own action"*. Later, Jagannathan and Wang [Jagannathan and Wang, 1996] recognized the mean-variance analysis and the Capital Asset Pricing Model as *"... major contributions of academic research to financial managers during the postwar era"*. Campbell and Viciera [Campbell and Viceira, 2002] wrote on Page 7, *"Most MBA courses, for exam-*

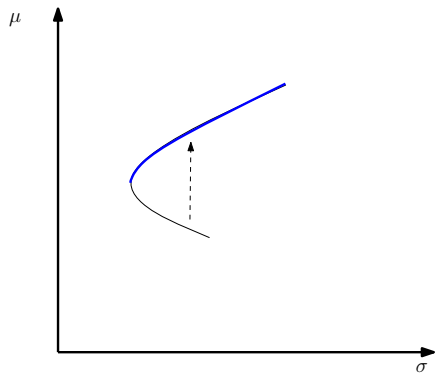


Figure 4.1: The illustration of the portfolio bound

ple, still teach mean-variance analysis as if it were a universally accepted framework for portfolio choice".

Portfolio theory is a theory of investment which tries to maximize the return and minimize the risk by carefully choosing different securities and is widely used in the finance industry. Security is a legal entitlement to receive (or an obligation to pay) an amount of money. A portfolio is a combination of existing securities, which tell us how many units of each security have to be bought or sold to create the portfolio [Černý, 2003]. The theory starts with relevant beliefs about future performances of available assets according to the past track records, and ends with a choice of portfolio. The portfolio theory models a portfolio as a weighted combination of securities. Let μ be the expected return and σ be the variance of the gross returns. The set of possible μ, σ-combinations offered by portfolios of risky securities that yield minimum variance for a given rate of return is called minimum-variance opportunity set or *portfolio bound*. It is illustrated in Figure 4.1. The upper part of the portfolio bound is called *efficient frontier*, also known as Markowitz frontier. All the portfolios on the efficient frontier have the highest attainable rate of return given a particular level of standard deviation. The efficient portfolios are candidates for the investor's optimal portfolio.

There are two salient features of any security investment.

- Uncertainty is an inherent feature of security investment. Economic forces are not understood well enough for predictions to be beyond doubt or error. The consequences of economic conditions are not understood perfectly. Moreover, non-economic influences, such as changes in international tensions, or a natural disaster, can change the success of a particular investment.

- The correlation among security returns is another inherent feature of security investments. As seen in the recent past, bank stocks were highly correlated. Or a country specific stocks could be correlated, *e.g.*, companies of Haiti. To reduce investment risk, it is necessary to avoid a portfolio whose securities are all highly correlated with each other. One hundred securities whose returns rise and fall in near unison afford little more protection than the uncertain return of a single security. The returns on correlated securities tend to move up and down together. Diversification of security investments could eliminate risk if their returns are not correlated.

The criteria for choosing an investment portfolio, which serves as a guide to the important and unimportant, the relevant and irrelevant, depends on the nature of the investor. Investors can be conservative, balanced, or aggressive based on their appetite for risk. Conservative investors may emphasize more on low risk. However, two objectives of portfolio analysis are common:

- Investors want the "return" to be high.

- They want this return to be dependable, stable, not subject to uncertainty.

By combining different securities whose returns are not correlated, portfolio theory seeks to reduce the total variance of the portfolio. The appetite for risk determines the variance. Due to its excellent performance, this theory is widely used today. Besides financial instruments, some experts have applied the theory to other areas and disci-

plines. Portfolio theory has been applied to portfolios of projects [Hubbard, 2007]. In developmental economics, Conroy [Conroy, 1975] modeled the labor force in the economy using portfolio-theoretic methods to examine growth and variability in the labor force. In social psychology, the self-concept consisting of self attributes is modeled using portfolio theory [Chandra and Shadel, 2007]. The predictions based on this model have been confirmed in studies involving human subjects. Recently, Wang *et al.* adopted portfolio theory in information retrieval in [Wang and Zhu, 2009]. The ranking problem is formulated as a portfolio selection problem. That is, in document retrieval, a top-n ranked list (portfolio) of documents is selected as a whole, rather than ranking documents independently. The volatility (the change) of the documents' relevance can be reduced by diversification. The weights of rank positions are chosen according to the discount factors in [Järvelin and Kekäläinen, 2002]. By deriving an objective function considering both relevance and uncertainty according to portfolio theory, the proposed document ranking algorithm sequentially selects a document and optimizes the objective function. As a result, a right combination of top-n relevant documents are chosen. The theory brings improved text retrieval performance. It is different from our multimedia portfolio fusion method. The weights are fixed for the retrieval work, and the method is to select appropriate documents.

The multimedia fusion problem is quite similar to portfolio analysis in finance. Each information source can be considered as a security in financial investment. The information source also has two salient features:

- First of all, the information source has the uncertainty feature. There is still no perfect learning to predict without doubt or error. For example, some objects may be misclassified in object detection if they have similar color with background.
- Moreover, it is quite common that some information sources are correlated. For example, two spatially proximate cameras will report results in near unison.

In multimedia fusion, we study each information source to obtain classifiers and invest different weights on each information source to obtain good classification results on future unseen instances. The objective is to achieve a high dependable return. Most of existing methods aim to maximize expected accuracy. However, it will not be able to guarantee actual high accuracy due to uncertainty. Even when we have a classifier with high expected accuracy, it is not safe if its variance is high [Breiman, 1996]. Take surveillance scenario as an example: camera information is generally reliable in human detection, *i.e.*, high expected detection accuracy. However, it works well in the day time, but can barely detect anything in the dark. That is, the performance varies dramatically from time to time, and thus the variance is also high. On the other hand, the audio sensor generally cannot perform detection as well as camera, but it can work in both the day time and the dark. That is, the audio sensor can have low expected accuracy as well as low variance. Thus, uncertainty is an extremely important feature that demands serious consideration. Moreover, the information sources in the multimedia systems are generally correlated. It is not always correct to assume independence of the information sources. Thus, diversification is beneficial for multimedia fusion. Poh *et al.* in [Poh and Bengio, 2005] discussed how the correlations affect the fusion performance. It is shown that the more dependent the information sources are, the lesser the gain one can benefit out of fusion. The positive correlation "hurts" fusion (fusing two correlated information sources of similar performance will not always beneficial) while negative correlation (greater "diversity") improves fusion. For example, the spatially proximate cameras will report results in near unison and have positive correlation. They may dominate the decision even if their predictions are wrong, and thus hurt the fusion performance. On the other hand, camera and audio sensor in surveillance capture different information from visual and audio data and may complement each other. Thus, they have negative correlation and can improve the fusion performance. As stated above, both uncertainty and correlation should be considered in multimedia fusion. It

is not advisable to only maximize the expected performance in multimedia fusion. We should attempt to maximize expected return (desired performance, *e.g.*, accuracy) and minimize risk (the uncertainty in the decision) to achieve an overall good performance. The portfolio theory is key in helping achieve this.

4.2 Problem formulation

- $\mathcal{S}$ is a multimedia system designed for accomplishing a detection task D. The multimedia system $\mathcal{S}$ consists of $N \geq 1$ correlated information sources $M_1, M_2, ..., M_N$.

- For $1 \leq i \leq N$, let $I_i(\mathbf{X})$ be the prediction of the detection task D based on the individual ith information source on instance $\mathbf{X}$. It is obtained by employing a detector on the features extracted from M_i, and can be either probabilistic output (posterior probability estimations) or decision output (belief values or -1/+1 decision values). The final prediction I of $\mathcal{S}$ consisting of information sources $M_1, M_2, ..., M_N$ is modeled as:

$$I(\mathbf{X}) = \sum_{i=1}^{N} w_i I_i(\mathbf{X}) \tag{4.1}$$

  where, w_i is the normalized weight assigned to M_i. $0 \leq w_i \leq 1, \sum_{i=1}^{N} w_i = 1$. In the classification problem, the prediction is the output of likelihood for different classes.

- For $1 \leq i \leq N$, let $r_i(\mathbf{X})$ be the return of M_i at $\mathbf{X}$ individually, and R_i be the expected return of M_i, which is defined as $R_i = E[r_i]$. The return depends on the application.

- For $1 \leq i, j \leq N$, let $\Phi = [\Phi_{ij}]$ be the covariance matrix between information sources. The element Φ_{ij} is defined as $\Phi_{ij} = E[(r_i - E[r_i])(r_j - E[r_j])]$. It captures the correlations of different information sources.

Our aim is to find the optimized weights w_i so that the fusion prediction I achieves good performance (desirable results, *e.g.*, high accuracy or high average precision, for different applications). To solve this problem, the portfolio theory is employed to obtain suitable weights. The portfolio theory helps pick a portfolio of securities according to their return and risk. To adopt it in multimedia fusion, the return and risk of each information source need to be defined first.

4.2.1 Return and Risk

Each information source in the multimedia system is considered the equivalent of a security in financial investment. The definition can be varied to different applications according to their aims. For example, in the classification problem, since the aim of the classifier of the information source is to accurately predict the labels and the performance is evaluated using accuracy, the return should be positive if the prediction is correct and negative otherwise. For ith information source on instance $\mathbf{X}$, the return $r_i(\mathbf{X})$ is defined as Equation 4.2:

$$r_i(\mathbf{X}) = \begin{cases} 1 & \text{if } h_i(\mathbf{X}) = y(\mathbf{X}) \\ -1 & \text{otherwise} \end{cases} \tag{4.2}$$

where $h_i(\mathbf{X})$ is the predicted class of ith information source on instance $\mathbf{X}$, and $y(\mathbf{X})$ is the actual class of instance $\mathbf{X}$. For the retrieval problem, where we evaluate the performance using average precision, the definition of return can be Equation 4.3:

$$r_i(\mathbf{X}) = \begin{cases} I_i(\mathbf{X}) - 0.5 & \text{if } y(\mathbf{X}) = 1 \\ -(I_i(\mathbf{X}) - 0.5) & \text{otherwise} \end{cases} \tag{4.3}$$

Based on this definition, the expected return of ith information source R_i is approximated using $r_i(\mathbf{X})$ over all the previous instances $\mathbf{X}_\alpha, \alpha = 1, \ldots, n$:

$$R_i = E[r_i] = \frac{1}{T} \sum_{\alpha=1}^{n} r_i(\mathbf{X}_\alpha) \tag{4.4}$$

The risk of information source is modeled as the standard deviation σ of return. For information source M_i,

$$\sigma_i^2 = E[(r_i - E[r_i])^2] \tag{4.5}$$

$\sigma_i \in [0, 1]$, and larger value indicates more risk.

4.2.2 Correlation

The correlation among different information sources represents how they co-vary with each other. In many situations, the correlation between information sources provides useful information. Moreover, diversification which is related to correlation among information sources is beneficial for multimedia fusion to reduce risk.

The popular Pearson's correlation coefficient is used to measure the correlation between different information sources. Pearson's correlation coefficient between two variables is defined as the covariance of the two variables divided by the product of their standard deviations. For information source M_i and M_j, the correlation ρ_{ij} is defined as Equation 4.6:

$$\begin{aligned} \rho_{ij} &= \frac{E[(r_i - E[r_i])(r_j - E[r_j])]}{\sigma_i \sigma_j} \\ &= \begin{cases} 1 & \text{if } i = j \\ \dfrac{\sum_{\alpha=1}^{n} (r_i(\mathbf{X}_\alpha) - R_i)(r_j(\mathbf{X}_\alpha) - R_j)}{\sqrt{\sum_{\alpha=1}^{n} (r_i(\mathbf{X}_\alpha) - R_i)^2} \sqrt{\sum_{\alpha=1}^{n} (r_j(\mathbf{X}_\alpha) - R_j)^2}} & \text{otherwise} \end{cases} \end{aligned} \tag{4.6}$$

$\rho_{ij} \in [-1, 1]$. The covariance matrix for N information sources is $\Phi = [\Phi_{ij}]_{N \times N}$, in which $\Phi_{ij} = \rho_{ij} \sigma_i \sigma_j$. The σ_i and σ_j are the standard deviation of returns for information source M_i and M_j, which is defined in Equation 4.5. $\Phi = [\Phi_{ij}]_{N \times N}$ captures the correlations and risk of multiple information sources.

4.2.3 Optimal Weights with Portfolio Theory

According to portfolio theory, the optimal portfolio is on the Markowitz frontier and it can be found by minimizing the expression 4.7, which is to maximize the return while minimize the variance of return:

$$f = \mathbf{w}^{\mathrm{T}} \Phi \mathbf{w} - \lambda \mathbf{R}^{\mathrm{T}} \mathbf{w} \tag{4.7}$$

where,

- $\mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix}$ is the vector of the weights for information sources. $0 \leq w_i \leq 1$ and $\sum_{i=1}^{N} w_i = 1$.
- $\mathbf{R} = \begin{bmatrix} R_1 \\ R_2 \\ \vdots \\ R_N \end{bmatrix}$ is the vector of the expected returns for information sources. $\mathbf{R}^{\mathrm{T}}\mathbf{w}$ models the expected return of the portfolio of information sources.
- Φ is the covariance matrix for the information sources in the multimedia system. $\mathbf{w}^{\mathrm{T}}\Phi\mathbf{w}$ models the variance of return of the portfolio.
- $\lambda \in [0, +\infty)$ is "risk tolerance" factor. Different values can be used for different "risk appetite" in various applications. $\lambda = 0$ results in minimizing the risk (conservative risk appetite), while $\lambda = +\infty$ results in maximizing the expected return of the fusion results (aggressive risk appetite).

To solve the optimization problem, we used the *quadratic programming* (QP) approach. A linearly constrained optimization problem with a quadratic objective function is called a quadratic program (QP).

In the multimedia portfolio fusion method, the aim is to minimize:

$$f = \mathbf{w}^{\mathrm{T}}\Phi\mathbf{w} - \lambda\mathbf{R}^{\mathrm{T}}\mathbf{w}$$

It is equivalent to minimizing (by multiplying $\frac{1}{2}$ on both sides)

$$f_{\mathbf{w}} = \frac{1}{2}f = \frac{1}{2}\mathbf{w}^{\mathrm{T}}\Phi\mathbf{w} + \mathbf{R}_p^{\mathrm{T}}\mathbf{w}$$

where, $\mathbf{R}_p = (-\frac{\lambda}{2}) \times \mathbf{R}$. There is one equality constraint ($\sum_{i=1}^{N} w_i = 1$):

$$\begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix} = 1$$

There are N inequality constraints ($w_i \geq 0$ for $i = 1, ..., N$):

$$\underbrace{\begin{bmatrix} 0 & 0 & \cdots & 1 & \cdots & 0 \end{bmatrix}}_{i} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix} \geq 0$$

In our case, Φ is the covariance matrix according to our definition. It can be proved that the covariance matrix is positive semidefinite. The problem is thus a convex QP and can be solved using the active set method. Active set method can be described in general as follows in Algorithm 4. The details of the algorithm can be found on page 472 of [Nocedal and Wright, 2006].

Generally, the "risk tolerance" factor λ is set such as the values of covariance Φ and return $\mathbf{R}_p$ are of the same order of magnitude to trade-off between risk and return. Too large λ will hurt the performance. Here, we simply set $\lambda = 1$, which corresponds to a moderate appetite for risk of a balanced investor.

Input: A quadratic program
Output: Feasible optimization solution
Compute a feasible starting point;
Set a subset of the active constraints at this point;
while *There are negative Lagrange multipliers* **do**
    solve the equality problem defined by the active set;
    compute the Lagrange multipliers of the active set;
    remove a subset of the constraints with negative Lagrange multipliers;
    search for infeasible constraints and update;
end

Algorithm 4: Active Set Method

4.2.4 Multimedia Portfolio Fusion

In summary, the portfolio fusion method is described in Algorithm 5 and 6, and is illustrated in Figure 4.2. The return and risk for information sources are first computed using the past (training) data. Then, portfolio theory gives the optimal weights for the information sources by minimizing the risk while maximizing the return, as shown in Algorithm 5. After that, as shown in Algorithm 6, for each test instance, the predictions from the information sources will be combined linearly using the optimal weights, and the class with largest prediction value will be chosen as the class of the instance. When a new information source is introduced, only the correlations will be computed against training the fusion model again in training-based fusion method, which saves many computation efforts.

4.3 Simulation Experiment

To show the effectiveness of our proposed portfolio fusion method, we simulated different cases of multiple information sources and applied to different fusion methods. We performed our simulation on the fusion of N information sources. The notations are in the following:

- Let $M_1, M_2, ..., M_N$ represent the set of N information sources.

Input: Labeled observations for N information sources,
"risk tolerance" factor λ
Output: Optimal weights $\mathbf{w}$ for the fusion
`// Calculate the return for each information source`
foreach *Information source* M_i **do**
  Train a classification or decision model $model_i$ using the observations in information source M_i;
  Calculate the return $r_i(\mathbf{X})$ for each observation in information source M_i;
  Obtain the expected return R_i for information source M_i;
end
`// Calculate the return vector`
Obtain the vector of the returns for all the information sources $\mathbf{R} = \begin{bmatrix} R_1 \\ R_2 \\ \vdots \\ R_N \end{bmatrix}$;
`// Calculate the covariance matrix of information sources`
foreach *Information source pair* M_i, M_j **do**
  Calculate the covariance Φ_{ij} of information source M_i, M_j;
end
Obtain the covariance matrix $\Phi = \left[\Phi_{ij}\right]_{N \times N}$;
`// Portfolio theory optimization`
$\mathbf{R}_p = (-\frac{\lambda}{2}) \times \mathbf{R}$;
Minimize $f_{\mathbf{w}} = \frac{1}{2}\mathbf{w}^{\mathrm{T}}\Phi\mathbf{w} + \mathbf{R}_p^{\mathrm{T}}\mathbf{w}$ using the constraints to obtain optimal weights $\mathbf{w}$ (*e.g.*, using Active Set Method);

Algorithm 5: Optimal Weights Determination by Portfolio Theory

Input: Optimal weights $\mathbf{w}$ for different information sources
Output: Prediction result c
In the future, for instance $\mathbf{X}$;
Calculate the prediction $I(\mathbf{X}) = \sum_{i=1}^{N} w_i I_i(\mathbf{X})$ for each class;
The class c with largest prediction value is the predicted result for $\mathbf{X}$;

Algorithm 6: Portfolio Fusion Method

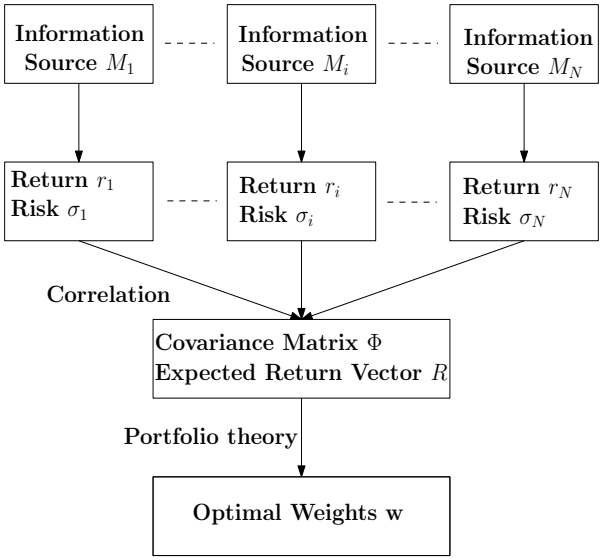


Figure 4.2: The architecture of the portfolio fusion method

- Let $\{M\}_1^K, K = 1, 2, ..., N$ represents the fusion result of information sources 1 to K.

- Let $\mathcal{N}(\mu, \sigma)$ represent a Gaussian distribution with mean μ and standard deviation σ.

We compared our fusion method with other widely used methods under different conditions. One is the weighted linear fusion. The weights are obtained according to the widely used Adaboost [Polikar et al., 2001] method. For information source M_i, $w_i = \frac{1}{2}\ln(\frac{1-e_i}{e_i})$, where e_i is the error rate of the classifier for information source i. The fusion result for N information sources is defined as:

$$I(\mathbf{X}) = \sum_{i=1}^{N} w_i I_i(\mathbf{X}) \tag{4.8}$$

We refer to this method as the *weighted fusion method*. We refer to the method that assigns the same equal weights to all different information sources as the *average fusion method*.

The other method is the naive Bayesian fusion [Manyika and Durrant-Whyte, 1994] method. The fusion results for N information sources is defined as:

$$I(\mathbf{X}) = \prod_{i=1}^{N} I_i(\mathbf{X}) \tag{4.9}$$

where $I_i(\mathbf{X})$ is the probabilistic output of information source M_i. We refer to this method as the *Bayesian fusion method*.

We now describe the results of the simulation runs. Section 4.3.1 gives the description of the simulations. Section 4.3.2 specifies what parameter changes are investigated in our simulation experiments and how they affect the results.

4.3.1 Simulation Setup

We performed our simulation on the fusion of N different information sources. We simulated N information sources $M_1, M_2, ..., M_N$ and then fused them one by one. For each information source, we generally assume two Gaussians for the negative and positive class [Aly and Hiemstra, 2009]. We randomly generate the annotated collection for each information source (which carries -1/+1 label for each instance). The negative class has distribution $\mathcal{N}(\mu_0, \sigma_0)$ and positive class has distribution $\mathcal{N}(\mu_1, \sigma_1)$. The number of instances is $SP = SN = 200$. To reduce the effects of randomness in the results, we repeated every simulation run $L = 50$ times. The results of each simulation run is actually obtained as an average over 50 times. The actual simulation process is described in Algorithm 7.

4.3.2 Simulation Parameter Variation

As our goal is to validate the performance of the proposed fusion algorithm, we vary the parameters combination to see the overall performance of the fusion algorithm and compare with other widely used methods. We choose different values for simulation parameters μ_0, σ_0, μ_1, σ_1 for each information source. Here, a large standard deviation

Input: $\mu_0, \sigma_0, \mu_1, \sigma_1$ for N information sources
Output: Fusion results for $\{M\}_1^K, K = 1, ..., N$
foreach *Repetition* $l \in [1..L]$ **do**
  foreach *Information source* M_i **do**
    `// Data generation`
    Generate SN negative samples from $\mathcal{N}(\mu_0, \sigma_0)$;
    Generate SP positive samples from $\mathcal{N}(\mu_1, \sigma_1)$;
    Select 50% negative and positive samples as training dataset;
    Use remaining negative and positive samples as test dataset;
    `// Model training`
    Train a classification model $model_i$ for information source M_i using the training data;
    Calculate the returns;
  end
  `// Modality fusion`
  foreach *Combination of information sources* $\{M\}_1^K$ **do**
    Find the optimal weights $\mathbf{w}$ using portfolio theory;
    Calculate classification performance and report the results;
  end
end
Report the average achieved performance over L repetitions;

Algorithm 7: The Simulation Procedure

represents more noise thus more uncertainty in the data. The large mean difference between positive and negative class $\mu_1 - \mu_0$ represents more discriminative in the data. The models are trained using LIBSVM [Chang and Lin, 2001]. For each simulation run, we use new seeds for the random number generator to ensure high quality of randomness. Here, we empirically set $\lambda = 1$.

In the simulation experiments, we tested the methods on six different scenarios with up to $N = 10$ information sources. The simulations are tested on both independent information sources and correlated information sources. Then, in each of these cases, we examined the information sources with different standard deviation and same standard deviation (both large discriminative and small discriminative information sources). The six scenarios are specified as follows:

1. Independent information sources with different standard deviation for each information source

2. Independent information sources with the same standard deviation for each information source but large difference between positive and negative class

3. Independent information sources with the same standard deviation for each information source but small difference between positive and negative class

4. Correlated information sources with different standard deviation for two categories of information sources

5. Correlated information sources with the same standard deviation for two categories of information sources but large difference between positive and negative class

6. Correlated information sources with the same standard deviation for two categories of information sources but small difference between positive and negative class

The simulation results are shown in Figure 4.3. The descriptions can be found in Table 4.1.

- Figure 4.3(a) and 4.3(d) show the simulation results of independent information sources with varying standard deviations (scenario 1). We set the standard deviation $\sigma_0 = \sigma_1 = i$ for $i = 1, ..., N$ with $\mu_1 - \mu_0 = 10$.

- The next two figures show the simulation results of independent information sources with the same standard deviation $\sigma_0 = \sigma_1 = 1$. Figure 4.3(b) and 4.3(e) show the experiment with larger mean difference between positive and negative class $\mu_1 - \mu_0 = 2$ (scenario 2), while Figure 4.3(c) and 4.3(f) show the experiment with smaller mean difference between positive and negative class $\mu_1 - \mu_0 = 1$ (scenario 3).

Scenario	**Correlation**	$\mu_1 - \mu_0$	σ
1	Independent	10	1–10 for each information source respectively
2	Independent	2	1
3	Independent	1	1
4	Correlated (information sources 1–9)	10	9 for correlated information sources, and 1 for the other one
5	Correlated (information sources 2–10)	2	1
6	Correlated (information sources 2–10)	1	1

Table 4.1: Descriptions of simulation scenarios: $\mu_1 - \mu_0$ denotes mean difference between positive and negative class, σ denotes standard deviation

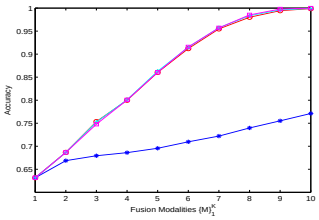
Then, the same scenarios are simulated on the fusion of correlated information sources. Here, for high correlation, we have 9 out of 10 information sources with the same data (the models are trained separately).

- In the simulation shown in Figure 4.3(g) and 4.3(j), we have $\sigma_0 = \sigma_1 = 1$ for one information source and $\sigma_0 = \sigma_1 = 9$ for the other 9 information sources (scenario 4). The mean difference between positive and negative class is $\mu_1 - \mu_0 = 10$.

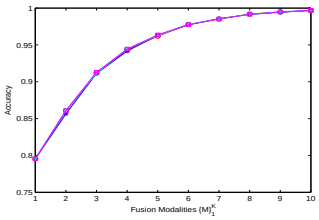
- In the other two simulations, we use information sources with the same standard deviation $\sigma_0 = \sigma_1 = 1$ but different mean difference between positive and negative class for different discrimination. The mean difference is $\mu_1 - \mu_0 = 2$ for Figure 4.3(h) and 4.3(k) (scenario 5), while the mean difference is $\mu_1 - \mu_0 = 1$ for Figure 4.3(i) and 4.3(l) (scenario 6).

We can draw the following conclusions from the simulation results,

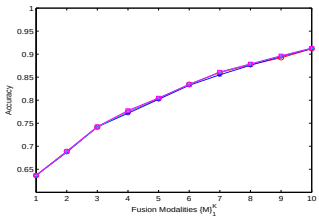
- For independent information sources (scenario 1-3),



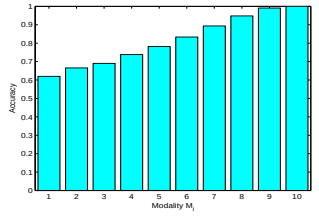
(a) Fusion results of scenario 1



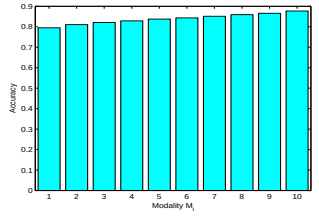
(b) Fusion results of scenario 2



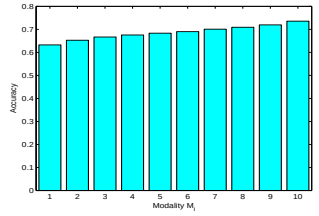
(c) Fusion results of scenario 3



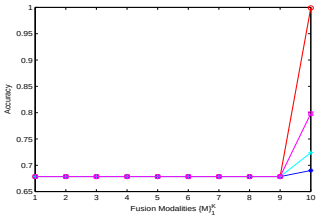
(d) Information source specification of scenario 1: independent information sources. $\sigma_0 = \sigma_1 = 11 - i$ for M_i with $\mu_1 - \mu_0 = 10$.



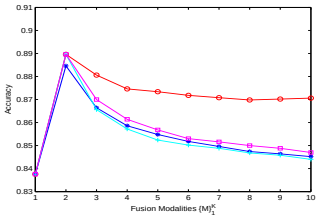
(e) Information source specification of scenario 2: independent information sources. $\sigma_0 = \sigma_1 = 1$ for all information sources with $\mu_1 - \mu_0 = 2$ for all information sources.



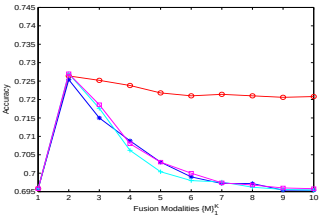
(f) Information source specification of scenario 3: independent information sources. $\sigma_0 = \sigma_1 = 1$ for all information sources with $\mu_1 - \mu_0 = 1$ for all information sources.



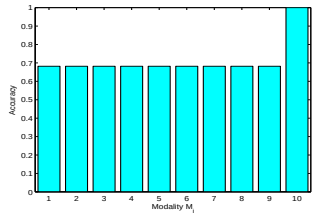
(g) Fusion results of scenario 4



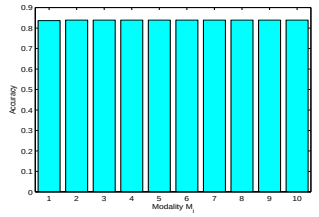
(h) Fusion results of scenario 5



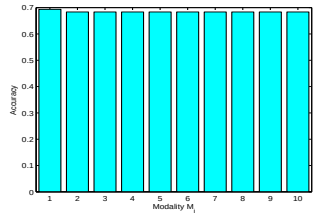
(i) Fusion results of scenario 6



(j) Information source specification of scenario 4: correlated information sources. $\sigma_0 = \sigma_1 = 9$ for the first 9 information sources, $\sigma_0 = \sigma_1 = 1$ for last with $\mu_1 - \mu_0 = 10$ for all.



(k) Information source specification of scenario 5: correlated information sources. $\sigma_0 = \sigma_1 = 1$ for all information sources with $\mu_1 - \mu_0 = 2$ for all information sources.



(l) Information source specification of scenario 6: correlated information sources. $\sigma_0 = \sigma_1 = 1$ for all information sources with $\mu_1 - \mu_0 = 1$ for all information sources.

Figure 4.3: The results of simulation runs for different simulation scenarios. Red circle represents the results of our proposed portfolio fusion method, blue asterisk represents the results of weighted fusion method, cyan plus sign represents the results of average fusion method and magenta square represents the results of Bayesian fusion method.

  - For the information sources that are of the similar performance (scenario 2-3), the fusion results are similar for all the 4 different fusion methods.
  - For the information sources that are of different performance (scenario 1), the fusion results of portfolio fusion, Bayesian fusion, and average fusion are similar, and all outperform the weighted fusion method by 2-20%.

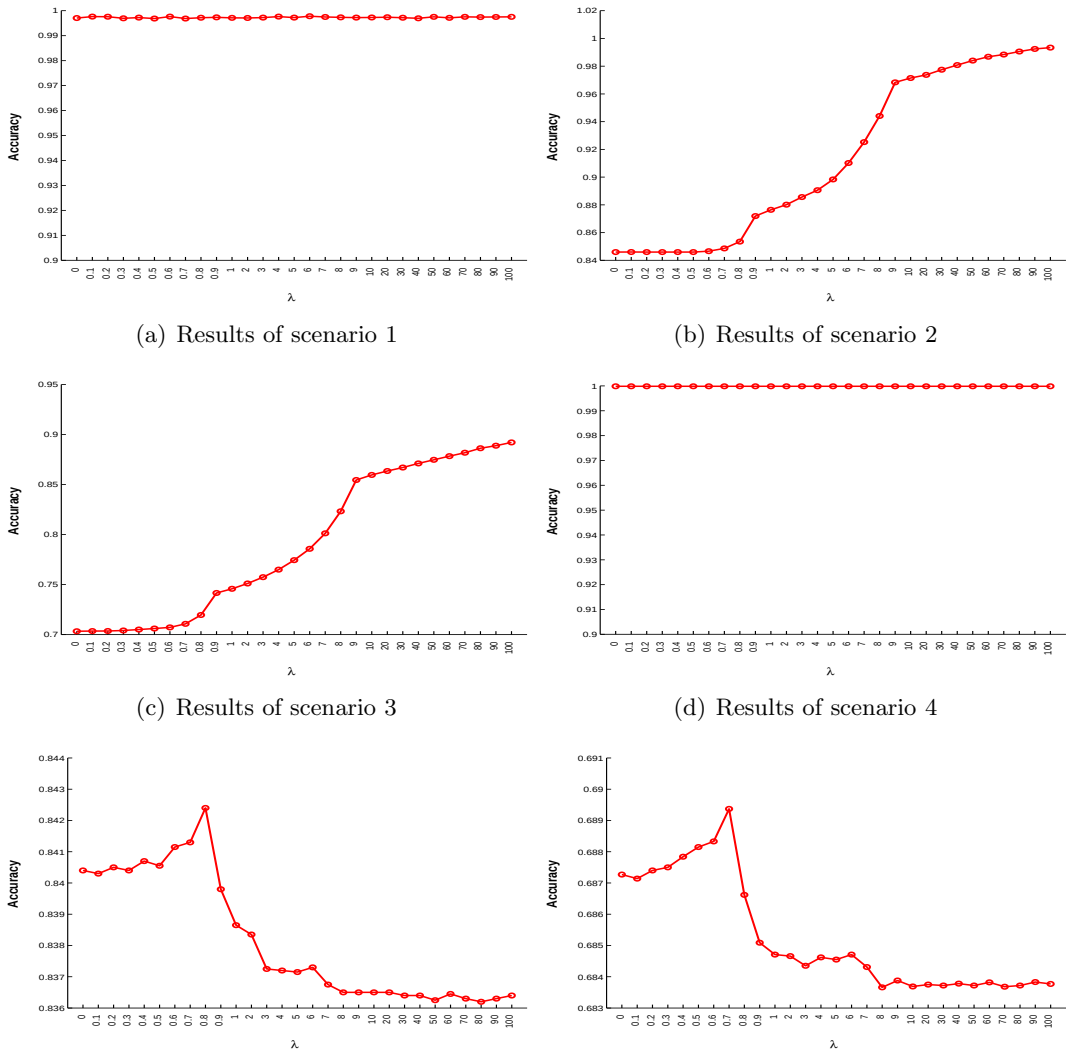
- For highly correlated information sources (scenario 4-6),
  - For the information sources that are of either similar (scenario 5-6) or different performance (scenario 4), the portfolio fusion method outperforms the Bayesian fusion, average fusion and weighted fusion methods. The portfolio fusion method outperforms the other fusion methods by about 3-15%.

It is observed that our portfolio fusion method is robust and has the best performance compared to the other methods in the various cases. Moreover, our portfolio fusion method can adapt to both probabilistic output and decision label output for the information sources. This is because the return and risk definition does not depend whether the output is probabilistic or deterministic. The Bayesian fusion method can only be used to probabilistic output.

4.3.3 Risk Tolerance Variation

We choose different values for risk tolerance and measure the performance in the 6 scenarios. The simulation results with different λ values (from very small value, *i.e.*, 0, to very large value , *i.e.*, 100) are shown in Figure 4.4.

It can be observed that for independent information sources scenarios, moderate values can achieve good performance. For correlated information sources scenarios (more emphasis on risk), small values can achieve better performance. In general, moderate values can always achieve rather good performance.



(a) Results of scenario 1

(b) Results of scenario 2

(c) Results of scenario 3

(d) Results of scenario 4

(e) Results of scenario 5

(f) Results of scenario 6

Figure 4.4: The results of simulation runs for different λ values.

4.4 Concept Detection Using Portfolio Fusion

To test the proposed portfolio fusion method for real applications, we evaluated it for concept detection on MSRA-MM dataset [Wang, Yang, and Hua, 2009]. There are 10,000 images labeled with respect to each concept. The dataset is equally divided into development and test sets: 5,000 images are selected for development of concept detection, and the other 5,000 images are for testing of concept detection. In the dataset, there are 50 concepts labeled non-exclusively for the images, such as mountain, ocean, indoor, building, cartoon.

In this experiment, four types of features from each images are exploited for the concept detection, including: (1) 64D HSV color histogram; (2) 256D RGB color histogram; (3) 75D edge distribution histogram; (4) 128D wavelet texture. The classification models are trained using the data from each information source with LIBSVM [Chang and Lin, 2001]. The attributes are scaled before applying SVM. When training SVM models, it is important to maintain balance between the number of positive and negative samples provided [Yanagawa et al., 2007]. In general, the concepts in the dataset are highly skewed towards negative samples (on average 6.5% positive samples). In our implementation, we utilized all available positive samples and randomly selected negative examples. The procedure for selecting negative samples is as follows: Let N_p denote the number of positive samples, and N_n denote the number of negative samples in the dataset. Take all the positive samples. Then,

- If $N_p > N_n$, all available negative samples are chosen.
- If $N_p < 10\% \times N_n$, we randomly selected 10% of negative samples.
- If $10\% \times N_n < N_p < N_n$, we randomly selected a set of negative samples equal in size to the number of positive samples.

The reliability of learned SVM models can also be highly sensitive to the selection of

model parameters. In our experiments, the objective is to evaluate the relative fusion performance rather than the absolute performance. Thus, we used the RBF kernel which in general is a reasonable first choice, and set the cost parameter to be 10. The other parameters were kept at default values [Chang and Lin, 2001].

After learning separate models for each feature, the outputs of each model are combined to obtain the fusion results. Here, we empirically set the "risk tolerance" factor $\lambda = 1$ (moderate risk) and compare our method with average fusion and Bayesian fusion methods which are popular and reported to have good performance for concept detection in [Li et al., 2009] and [Zheng et al., 2008].

The evaluation criteria for concept detection is the mean average precision (MAP), which is the mean of average precision (AP) for each concept. The AP is defined as:

$$AP = \frac{\sum_{k=1}^{K} P(k) \times R(k)}{T} \tag{4.10}$$

where k is the retrieved rank, K is the total number of images retrieved, $P(k)$ is the precision of retrieved first k images, $R(k)$ is the relevance of image at rank k (0 or 1), and T is the total number of relevant images in the corpus. Precision is defined as:

$$P = \frac{\#\text{retrieved relevant images}}{\#\text{retrieved images}} \tag{4.11}$$

Relevance $R(k)$ is defined as:

$$R(k) = \begin{cases} 0 & \text{if image k is not relevant} \\ 1 & \text{if image k is relevant} \end{cases} \tag{4.12}$$

The average precision for each concept is calculated over the retrieved relevant images $K = 5,000$ and is shown in Figure 4.5. Here, only the concepts with precision larger than 1×10^{-7} in any fusion method are shown. There are 14 concepts, including animal, building, cartoon, crowd, earth, indoor, man, mountain, ocean, outdoor, people, sky, vegetation, and woman. As shown in Figure 4.5, the portfolio fusion method outperforms Bayesian fusion on 13 concepts (except earth which has same performance) and

average fusion on 14 concepts. It suggests that our portfolio fusion method can make good use of correlation and uncertainty in decisions in different information sources. For example, in the concepts of mountain and ocean, the information sources have similar performances and are highly correlated. The information sources can be approximately considered as one group and the weights can be assigned to any source. As a result, the portfolio fusion method assigns all the weight to HSV color histogram feature. Because the information sources are highly correlated, the fusion does not improve the performance much and the average precision for different fusion methods are all low. In addition, the results of our portfolio fusion method are generally better for almost all the concepts. It indicates the superiority of our proposed portfolio fusion method against the average fusion method and Bayesian fusion method. It may be because our fusion method made better use of correlation and uncertainty of the decisions from different information sources. The correlation of different information sources in different

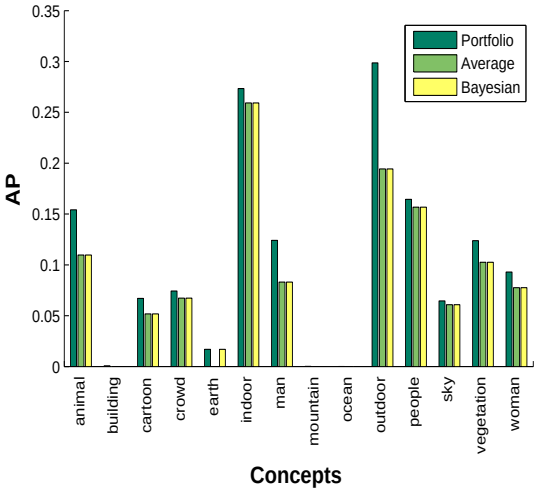


Figure 4.5: Average precision of each concept

concepts is shown in Figure 4.6. The whiter the image, the larger the correlation. It shows that the correlation is consistent with the performance. For the concepts that have diverse information sources and some are highly correlated, *e.g.*, outdoor and man,

the improvement is significant. But for the concepts that all the information sources are highly correlated or rather independent, *e.g.*, indoor and sky, the improvement is not so much.

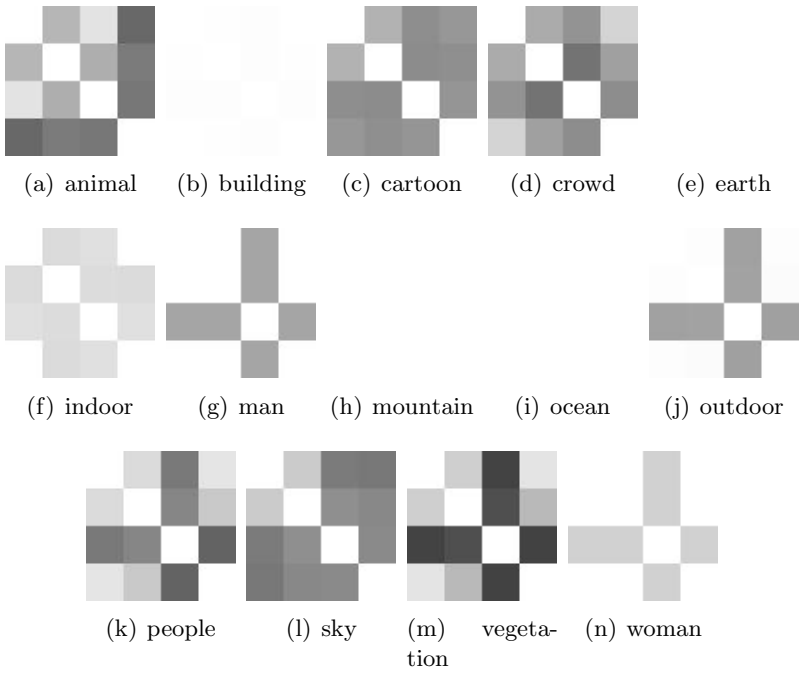


Figure 4.6: The correlation of different information sources in different concepts

The mean average precision (MAP) results are shown in table 4.2. It can be observed that the MAP of our portfolio fusion method outperforms MAP of the average fusion and Bayesian fusion by about 24% (relative).

4.5 Human Detection Using Portfolio Fusion

The portfolio fusion method is also evaluated for human detection. The dataset is recorded using multiple sensors. There are three audio sensors and two cameras. The sensor layout schema in Figure 4.7 shows the relative camera and audio sensor position and overlap. The example camera views are shown in Figure 4.8. Comparing to the dataset in Chapter 3, the dataset here is captured by multiple sensors (3 audio sensors

and 2 cameras).

Methods	Average fusion	Bayesian fusion	Portfolio fusion
MAP	0.083	0.084	0.104

Table 4.2: MAP by different fusion methods

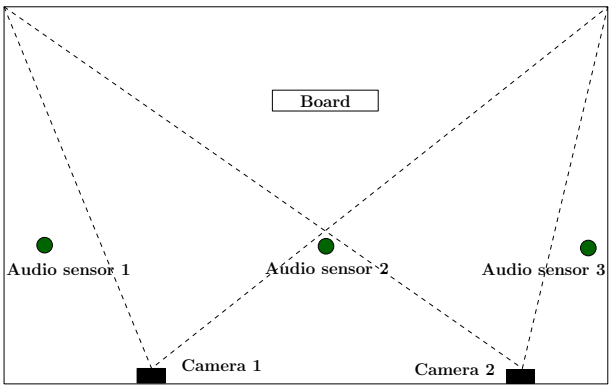


Figure 4.7: Sensor layout schema

The task is to detect whether there is human in the region. The data in different streams are first synchronized by timeline. Then, the data are segmented into frames and corresponding audio samples as the examples. There are 840 examples, each with one frame and corresponding 1 second audio samples. 420 examples are selected as training set, and the remaining examples are treated as test set. The features using for human detection are as follows:

- Audio: the audio energy. For each time interval, the audio energy can be easily calculated as the sum of squared audio samples.
- Visual: the frame difference with the background.

The features are easy to use and reasonable for human detection. Moreover, the objective is to demonstrate the proposed method works well compared to other fusion methods. Though not too complicated features are used, it should not impact the relative performance of the different fusion methods.

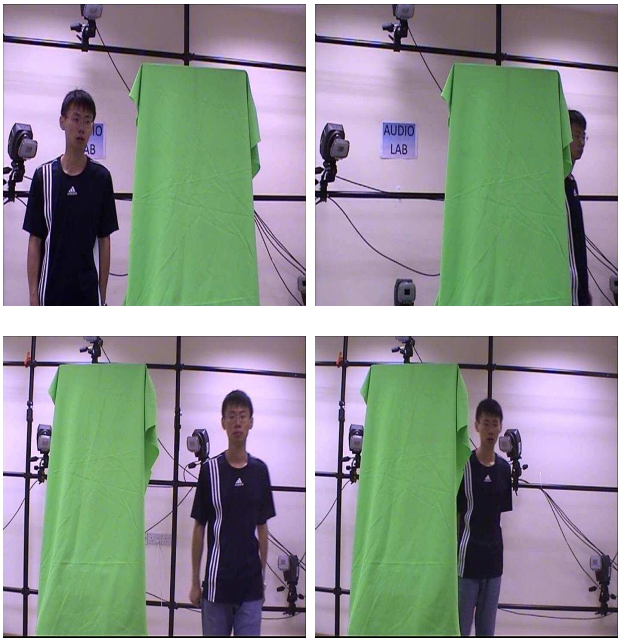


Figure 4.8: Example camera views

The model for each information source is trained using LIBSVM with default parameters. Then, the portfolio fusion method is compared with average fusion method, Bayesian fusion method, weighted fusion method, and MultiFusion method. The "risk tolerance" factor for portfolio fusion is empirically set to be $\lambda = 1$ here.

The performance of each information source is shown in Figure 4.9. Table 4.3 illustrates the fusion results with different fusion methods. The video and audio information sources are highly correlated, and degraded the fusion performance in other fusion method. The performance in other fusion methods are lower and almost the same. That is because the audio information sources are majority and dominate the performance in other fusion methods. The portfolio fusion method makes use of the correlation and uncertainty. It outperforms the other three methods by about 7.8% (relative). The correlation of different sensors in recorded data is shown in Figure 4.10. It shows that the audio sensors are highly correlated, and can dominate the results since they are in majority.

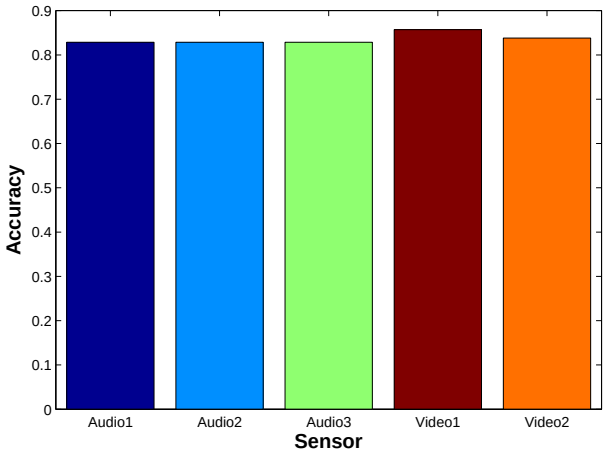


Figure 4.9: The performance of each information source

Figure 4.10: The correlation of different information sources for recorded data

Methods	Average	Bayesian	Weighted	MultiFusion	Portfolio
Accuracy	82.86%	82.86%	82.86%	82.86%	89.29%

Table 4.3: Detection accuracy by different fusion methods

The significance tests have been conducted. The experiments have been repeated 20 times and the *t-test*s have been done. It shows that the proposed portfolio fusion method passes the *t-test* with other fusion methods at the 5% significance level.

4.6 Conclusion

In this chapter, a novel multimedia fusion method using portfolio theory is proposed. It shows that the proper modeling of correlation among information sources can help to improve the performance. The proposed method can be applied to either probabilistic output or decision output. Moreover, it is easily scalable. Our proposed fusion method does not require additional learning for weights after models for each information source are trained. When a new information source is introduced, only the correlations will be computed instead of training the fusion model again. With well defined returns and risk, portfolio fusion method tries to maximize the return while minimizing the risk. Using appropriate definition of returns and risk, the method can also be adapted to different application scenarios. It is shown to achieve good performance in actual experiments. The proposed fusion method can be tuned for different risk appetite of applications by using proper risk tolerance values. More study will be done on exploiting recent advances in modern Portfolio theory, such as, dynamic correlations adaptation, for improving the performance.

Chapter 5

Up-Fusion

The first two chapters show that proper utilization of correlation (multiple utilization or sophisticated modeling) can improve the fusion performance. As discussed in Section 2.4 Chapter 2, another important issue is that the fusion model is generally not evolving. In multimedia fusion, the evolution of fusion models is of primary importance because of the nature of multimedia applications. First of all, the semantic label information is important for multimedia analysis because many multimedia analysis tasks are based on classification and a large amount of labeled training data are necessary for good classification. However, most of multimedia data have limited label information, or worse yet, have no label information. For example, on Flickr, the labels for the multimedia documents (images, tags and descriptions) are not available or quite noisy. Labeled examples are fairly expensive to obtain due to the high labor costs faced when annotating videos [Wang et al., 2007]. Thus, little amount of training data are available at the beginning. The fusion performance may suffer as a result. Furthermore, the multimedia data keep increasing with time. New instances of multimedia data are continuously added. For example, new video are periodically uploaded to Youtube. Thus, the fusion model may not always be valid or effective as the multimedia data increase because the nature of the data collection can change. As a result, it will be quite useful

to evolve the multimedia fusion models and improve the performance with new data.

Most of the traditional fusion methods are static with respect to time. To address this, in recent years, several evolving fusion methods have been proposed. However, they can only be used in limited scenarios. For example, the context-aware fusion methods need the context information to update the fusion model, but the context information may not always be available in many applications. In this chapter, a new evolving fusion method, called *Up-Fusion*, is proposed based on the online portfolio selection theory. The proposed method takes the correlation among different information sources into account, and evolves the fusion model when new multimedia data are added. It can deal with either crisp or soft decisions without requiring additional context information. Pseudo-labels are used in the case when the label information of newly added data are not available. A sliding window approach has been utilized to deal with temporal changes of the multimedia data. The key contributions are:

- The method evolves the fusion model along with the newly added multimedia data to improve the performance.
- The evolution of fusion method considers the correlation among different information sources, can deal with both crisp and soft decision, and no context information is required. The situations that the labels of newly added data are not available and that context or nature of data changes, are also solved with proper refinement.

The rest of this chapter is organized as follows. Section 5.1 briefly reviews the related fusion methods and the motivation for the evolving fusion method. Section 5.2 introduces the online portfolio selection theory and discusses why and how it can be useful for multimedia fusion. Section 5.3 describes the proposed Up-Fusion method. Section 5.4 gives some refinements on the method, including the pseudo labels and the sliding window. Experimental results on concept detection and human detection are shown in section 5.5. Section 5.6 concludes the chapter with a summary of the

proposed work and discussions.

5.1 Related Work

Most of the traditional decision fusion methods are static fusion methods, such as, min/max/average fusion, majority vote fusion, Bayesian fusion, weighted fusion, and super-kernel fusion, *etc.*. That is, the fusion models in the methods stay unchanged no matter how the nature of data varies. Generally speaking, the fusion rules of these methods are predefined or classification based. The correlation and the different performances of different sources are generally not considered. The information sources in the multimedia systems are generally correlated. For example, two spatially proximate cameras will usually capture similar images. It is not always correct to assume independence of the modalities. Poh *et al.* in [Poh and Bengio, 2005] discussed how the correlations affect the fusion performance. It is shown that the more dependent the information sources are, the lesser the gain one can benefit out of fusion. The positive correlation "hurts" fusion (fusing two correlated information sources of similar performance will not always beneficial) while negative correlation (greater "diversity") improves fusion. Based on this understanding, a fusion method based on the portfolio selection theory is proposed in [Wang and Kankanhalli, 2010b]. With the mean-variance analysis, the portfolio fusion finds the optimal fusion weights for different information sources by minimizing the correlation while maximizing the performance. But it is still a static method.

More importantly, once obtained, the fusion models in these fusion methods are static over time. In reality, the correlation and reliability of information sources might vary with the increase of data, or the changes of context. The static fusion methods cannot adapt to the changing data and environment, which may make the methods unreliable or even fail to work. Particularly, the portfolio fusion method [Wang and

Kankanhalli, 2010b] cannot be simply extended for evolution:

- First and foremost, the portfolio fusion method needs all the data labeled. But in many multimedia applications, correct labels of the new data are not available.
- Second, simply applying portfolio fusion cannot guarantee to improve the fusion performance and it is inefficient to update the fusion model whenever there is a new data instance.
- Third, the correlation and reliability of different information sources can vary over time. Simply considering all the previous data to update the fusion model is thus also not appropriate.

There are also a few evolving fusion methods, such as adaptive fusion method [Chen and Ansari, 1998], confidence evolution method [Atrey and Saddik, 2008], and context-aware fusion method [Movellan and Mineiro, 1998; Lee and Park, 2008; Geng et al., 2010].

- Chen *et al.* proposed an adaptive fusion method in [Chen and Ansari, 1998]. They modeled the decisions as conditional probabilities and used log-likelihood as weights for each information source. The weights are updated according to their agreements with the fusion decision at each iteration. However, only crisp decisions are considered in the method, and it is not consistent with Principle of Least Commitment. Therefore, the possible hypotheses are dropped intermediately and the performance may be degraded. For clarity, we refer it as *crisp decision fusion method*.
- A confidence evolution method is described in [Atrey and Saddik, 2008]. The method needs training for initial confidence for individual information source. Then, at each instance, the information sources are divided into two subsets based

on their decisions. The confidences are updated according to their agreement coefficients with the subsets. The methods need trusted information sources. Moreover, only confidence is updated according to the agreement coefficients. The fusion model is based on the underlying assumption that the media streams are independent, and the correlation among information sources are not considered. The method needs to update the confidence for each new instance. It is inefficient, and a significant restriction is that the labels may not be available online, as it may require manual intervention at every update step. A more realistic scenario is the update of the existing fusion model when a new batch of data becomes available.

- Recently, some context aware fusion methods have been proposed like [Movellan and Mineiro, 1998], [Lee and Park, 2008], [Geng et al., 2010]. In *context weight fusion method* [Lee and Park, 2008], adaptive weighting scheme is adopted for acoustic and visual speech recognition. The weights for audio and visual vary according to the noise level in speech. The method needs the context information which may not be available and dealing with all influential context factors is unrealistic. Again, correlation among information sources is not considered. The correlation is an important factor for multimedia fusion. Proper utilization of correlation among different information sources can improve the fusion performance [Wang and Kankanhalli, 2010b].

In this chapter, we propose an evolving fusion method based on the online portfolio selection theory.

- Compared to the previous static fusion methods, especially portfolio fusion method, our proposed fusion method is evolutionary. The fusion model evolves as new data being added. In this way, the fusion model can adapt to the changing data and environment conditions. Suitable fusion models for different conditions should

improve the performance than a fixed fusion model.

- Compared to the previous evolving fusion methods, our proposed method utilizes the correlation among different information sources, can deal with either crisp or soft decision, and no context information is required. By taking correlations into account, a more proper fusion model can be achieved. Moreover, by dealing with different decisions without requiring the context information, the fusion method can be employed in various application scenarios. Even in the situation where the context information is available, the data in the same context situation can be measured to update the fusion model for this context situation. In this way, our fusion method should further improve the performance.

5.2 Online Portfolio Selection

In the dynamic multimedia application scenarios, the multimedia fusion method should be able to improve the fusion performance as the amount of available data increase. Moreover, the correlations among different information sources should be considered to achieve an appropriate fusion model. The *online portfolio selection theory* is appropriate for these requirements. First of all, the online portfolio selection theory considers the correlations among different stocks for investment. Second, the online portfolio selection theory gets the previous prices and updates the investment accordingly for better performance.

Online portfolio selection [Helmbold et al., 1998] is a mechanism developed in economics and finance. Consider a portfolio containing n stocks. Each trading day, the performance of the stocks can be described by a vector of price relatives, denoted by $\mathbf{x} = \{x_1, x_2, \ldots, x_n\}$, where x_i is the next day's opening price of the ith stock divided by its opening price on the current day. Thus the value of an investment in stock i increases (or falls) to x_i times its previous value from one morning to the next. A portfolio is

defined by a weight vector $\mathbf{w} = \{w_1, w_2, \ldots, w_n\}$ such that $w_i \geq 0$ and $\sum_{i=1}^{n} w_i = 1$. The ith entry of a portfolio $\mathbf{w}$ is the proportion of the total portfolio value invested in the ith stock. The online portfolio selection strategy is as follows: At the start of each day t, the portfolio selection strategy gets the previous price relatives of the stock market $\mathbf{x}^1, \ldots, \mathbf{x}^{t-1}$. From this information, the strategy immediately selects its portfolio $\mathbf{w}^t$ for the day. Over time, a sequence of daily price relatives $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^T$ is observed and a sequence of portfolios $\mathbf{w}^1, \mathbf{w}^2, \ldots, \mathbf{w}^T$ is selected.

The mechanism aims to maximize the wealth on each day based on previous observations. Similarly, we want to improve the multimedia fusion performance as the data increasing in multimedia systems. The major difference is that everyday we can observe the price and the return in the stock investment. In our multimedia fusion, the scenario is similar if the "correct" labels of the new instances can be revealed for each update. Unfortunately, it is a challenging task because the multimedia data are generally provided without labels and there is no perfect classification model that can always give correct labels. The availability of correct labels of the new instances is not possible in many situations. Thus, we will also consider the case that the labels for the new instances are not available. When there are multiple information sources, different information sources generally make mistakes on different instances. Thus, intuitively we can get nearly perfect correct labels with multimedia fusion.

5.3 Up-Fusion Method

- $\mathcal{S}$ is a multimedia system designed for performing a task D, such as retrieval or classification. The multimedia system $\mathcal{S}$ consists of $N \geq 1$ correlated information sources $M_1, M_2, \ldots, M_N$.

- For $1 \leq i \leq N$, let $I_i(\mathbf{X}) \in [0, 1]$ be the decision of the task D based on the ith information source on instance $\mathbf{X}$. It is usually obtained by employing a detector

on the features extracted from information source i. The final prediction I of $\mathcal{S}$ is modeled as the fusion of $I_i(\mathbf{X}), i = 1, 2, \ldots, N$ based on the fusion model.

- For $1 \leq i \leq N$, let $r_i(\mathbf{X})$ be the return of information source M_i at $\mathbf{X}$, and R_i be the expected return of information source M_i, which is expressed as $R_i = E[r_i]$. The return depends on the requirement of different applications. More specifically, $r_{i;X_{\alpha:\beta}}$ denotes the returns for instances $\mathbf{X}_\alpha$ to $\mathbf{X}_\beta$ based on information source M_i.

- For $1 \leq i, j \leq N$, let $\Phi = [\Phi_{ij}]$ be the covariance matrix of information sources. The element Φ_{ij} is defined as $\Phi_{ij} = E[(r_i - E[r_i])(r_j - E[r_j])]$. It captures the correlation of different information sources.

- For $0 \leq t \leq T$, let f_i^t be the model of information source M_i at iteration t, and F^t be the multimedia fusion model obtained at iteration t. For example, there are many video cameras in multimedia surveillance systems. Each camera is an information source, and the system will have a classification model to detect certain event for each camera. At time t, the classification model of camera i is f_i^t. The fusion model for the surveillance system is F^t, which is a combination of the models for different cameras f_1^t, ..., f_N^t.

- Y is a set of classes. $y(\mathbf{X})$ is the true label of instance $\mathbf{X}$.

Some of the symbols are summarized in Table 5.1. The fusion flow is described in Algorithm 8, and the procedure can be illustrated in Figure 5.1.

5.3.1 Definition

Each information source in the multimedia system is considered equivalent to a security in financial investment. The definition of return, expected return, risk of information

Symbol	Meaning
M_i	ith information source
$r_i(\mathbf{X})$	return of ith information source at $\mathbf{X}$
ρ_{ij}	correlation between information source i and j
f_i^t	the model of information source i at iteration t
F^t	fusion model at iteration t
$\mathbf{w}^t$	weights for different information sources at iteration t
λ	"risk tolerance" factor

Table 5.1: Summary of used symbols

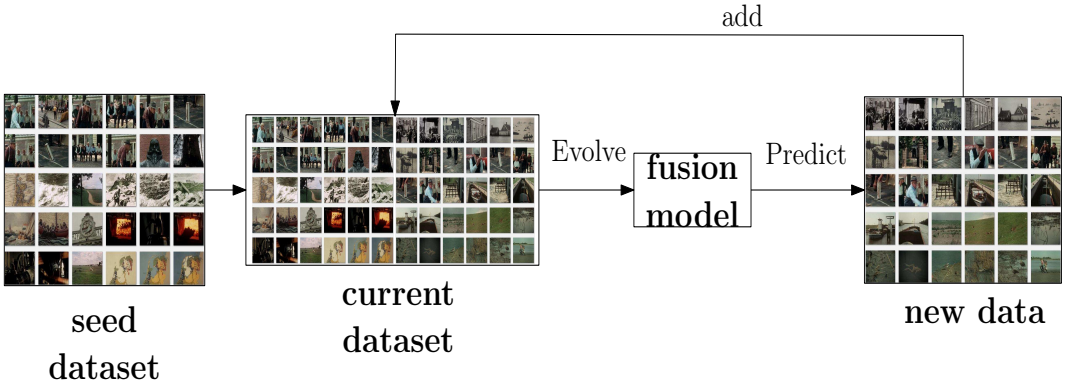


Figure 5.1: The framework of the proposed Up-Fusion method

source, correlation between information sources, and covariance matrix can be found in Chapter 4 Section 4.2.

With the portfolio fusion method, the optimal weights $\mathbf{w}$ for different information sources are obtained by minimizing:

$$\varphi = \mathbf{w}^{\mathrm{T}}\Phi\mathbf{w} - \lambda\mathbf{R}^{\mathrm{T}}\mathbf{w} \tag{5.1}$$

Here, $\mathbf{w}^{\mathrm{T}}\Phi\mathbf{w}$ is the variance (risk) of the information sources. $\mathbf{R}^{\mathrm{T}}\mathbf{w}$ is the return. $\lambda \in [0, +\infty)$ is a "risk tolerance" factor. The formulation is to maximize the return while minimizing the risk.

5.3.2 Initialization

The method starts with a dataset of N_0 labeled instances. This dataset is called the *seed dataset*. The classification model for individual information source can be obtained

Input: Seed dataset (the initial labeled dataset)

- Initialization (Section 5.3.2)
  - With the seed dataset, the classification model f_i for individual information source can be obtained
  - The return $\mathbf{R}^0$, as well as the covariance matrix Φ^0 for the information sources can be obtained according to Equation (5.2) and (5.3) based on the seed dataset
  - The initial fusion model F^0 is constructed using the portfolio fusion based on the expectation $\mathbf{R}^0$ and correlation Φ^0 obtained from seed dataset using Equation (5.4)
- Evolution (Section 5.3.3)
  - At each iteration t, K new instances are added. The decisions can be obtained using the previous portfolio fusion model F^{t-1}
  - Consequently, the expectation $\mathbf{R}^t$ and correlation Φ^t for the information sources will be updated using Equation (5.5) and (5.7). The portfolio fusion model F^t will thus be updated according to Equation (5.8)

Output: Fusion model F^t

Algorithm 8: Proposed Up-Fusion Method

with the labeled data. Here, binary classification is considered because multi-class classification can be achieved by the One-Versus-the-Rest strategy. The classification model for information source i is denoted as f_i^0. The decision according to f_i^0 on instance $\mathbf{X}$ is $I_i(\mathbf{X})$.

With the initial dataset, the expected return $\mathbf{R}^0$ and covariance Φ^0 are calculated. The initial expected return is

$$\mathbf{R}^0 = [R_i^0]_{N\times 1} \tag{5.2}$$

The initial covariance matrix for n information sources is $\Phi^0 = [\Phi_{ij}^0]_{N\times N}$, in which

$$\Phi_{ij}^0 = \rho_{ij}^0 \sigma_i^0 \sigma_j^0 \tag{5.3}$$

The optimal weights $\mathbf{w}^0$ for each information source are obtained by minimizing

$$\varphi = (\mathbf{w}^0)^{\mathrm{T}} \Phi^0 (\mathbf{w}^0) - \lambda (\mathbf{R}^0)^{\mathrm{T}} (\mathbf{w}^0) \tag{5.4}$$

The initial fusion model is $F^0 = \mathbf{w}^0 \cdot f_i^0$.

5.3.3 Evolution

The fusion model is updated at every iteration when new data are added. It will be inefficient to update the fusion model whenever there is a new data instance. Moreover, a significant constraint is that the labels will not be discovered soon after the prediction is made. In our Up-Fusion method, we will update the fusion model when a batch of K new instances becomes available. At iteration $t(t = 1, 2, \ldots, T)$, K new instances are added into the dataset and the data instances are $\mathbf{X}_{1:N_t}$.

According to the definition, the return $\mathbf{R}^t = [R_i^t]_{N\times 1}$, in which R_i^t is defined as:

$$R_i^t = E[r_{i;X_{\alpha_t:\beta_t}}] \tag{5.5}$$

The correlation ρ_{ij}^t between information source i and j can be updated as follows:

$$\rho_{ij}^t = \frac{E[(r_{i;X_{\alpha_t:\beta_t}} - R_i^t)(r_{j;X_{\alpha_t:\beta_t}} - R_j^t)]}{\sigma_{i;X_{\alpha_t:\beta_t}}\sigma_{j;X_{\alpha_t:\beta_t}}} \tag{5.6}$$

Thus,

$$\Phi_{ij}^t = \rho_{ij}^t\sigma_i^t\sigma_j^t = E[(r_{i;X_{\alpha_t:\beta_t}} - R_i^t)(r_{j;X_{\alpha_t:\beta_t}} - R_j^t)] \tag{5.7}$$

For the first step, the exact return and covariance method is used. That is, take all the current available data instances $\mathbf{X}_{1:N_t}$ into account, and calculate the return on the instance with Equation (4.2) or (4.3). Then, the new R_i^t and Φ^t are re-calculated on the whole dataset based on the definition. New R_i^t is calculated using Equation (5.5), and Φ^t is calculated using Equation (5.7). Here, $\alpha_t = 1$ and $\beta_t = N_t$.

The distribution of the newly added data instances may be largely different from the actual distribution, or the correlation of the different information sources on the newly added data instances varies from the actual correlation. The noisy new data instances may degrade the fusion performance. Thus, merely computing the exact return and covariance may not always improve the results. The performance may be unstable

and go up and down as the data increasing. In order to overcome this disadvantage, we refine the evolving fusion method by introducing a validation step. When new data instances are added, the weights can be obtained by the Up-Fusion method. Then, the weights are validated on the initial seed dataset. If the performance on the initial seed dataset is improved compared to the previous weights, the new weights are updated. Otherwise, the weights remain unchanged. In this way, we can expect the fusion performance to be always improved.

Thus, the weights $\mathbf{w}^t$ for each information source at iteration t are obtained by minimizing

$$\varphi = (\mathbf{w}^t)^{\mathrm{T}} \Phi^t(\mathbf{w}^t) - \lambda(\mathbf{R}^t)^{\mathrm{T}}(\mathbf{w}^t) \tag{5.8}$$

Subject to:

- $\sum_{i=1}^{N} w_i^t = 1$, and $0 \leq w_i^t \leq 1$
- $\mathcal{P}(\mathbf{w}^t) \geq \mathcal{P}(\mathbf{w}^{t-1})$. Here, $\mathcal{P}(\mathbf{w})$ denotes the fusion performance on seed dataset with weights $\mathbf{w}$

To take the prior knowledge into account, the initial point for minimization is set to be the previous weights. Starting from the initial weight vector, the formula is optimized as a quadratic programming problem. If the performance on validation dataset with new weights is better than that of the old ones, the fusion model is updated with new weights. Otherwise, the weights remain unchanged. In this way, the method evolves the fusion model to improve the fusion performance. The fusion model at iteration t is then expressed as: $F^t = \mathbf{w}^t \cdot f_i^t$. Here, $f_i^t = f_i^0$ because the classification model is not re-trained when new data instances are added. The Up-Fusion method only updates the correlation of different information sources at each iteration. The computational complexity is $O(N^2 d)$, where N is the number of information sources, d is the number of data instances for update, and usually $N \ll d$. The optimal weights

can be found in polynomial time of N (usually $O(N^2)$). Thus, the total complexity for each iteration is $O(N^2 d)$.

The evolution is one of our key contributions. Compared to static fusion method, the fusion model is updated at every iteration when new data are added. Compared to the previous evolving fusion methods, the evolution utilizes the correlation among different information sources, can deal with either crisp or soft decision, and no context information is required.

5.4 Refinement

5.4.1 Pseudo labels

With the above procedure, the baseline version of Up-Fusion model can be obtained. However, the calculation of correlation and return needs the true labels of the newly added data instances. There are also many situations that the true labels may be not, or costly, available. In order to incorporate the Up-Fusion method with the situation that the true labels of the newly added data are unknown, the pseudo labels are introduced. Instead of the true labels, the predicted labels on the newly added data according to the previously obtained fusion model are used as the labels of the new data. Specifically, at a time, a fusion model F is achieved. For a data instance $\mathbf{X}$ whose true label is unknown, the pseudo label $y^\star$ is defined as:

$$y^\star(\mathbf{X}) = \arg\max_{y \in \mathcal{Y}} F_y(\mathbf{X}) \tag{5.9}$$

where $\mathcal{Y}$ is the set of labels, and F_y is the confidence for class y with fusion model F. That is, the most probable label of $\mathbf{X}$ based on the current fusion model is considered as the pseudo label. The pseudo label $y^\star(\mathbf{X})$ is then used as $y(\mathbf{X})$ to calculate the return and correlation. This strategy is similar to the co-training method in semi-supervised learning algorithm [Blum and Mitchell, 1998]. The co-training approach [Blum and

Mitchell, 1998] has proven to converge, if two assumptions hold:

- (a) the error rate of each classifier is low
- (b) the views must be conditionally independent

For practical usage, co-training can even be applied, if the learners are slightly correlated. In the Up-Fusion method, the two assumptions generally also hold:

- First, it is reasonable to assume decent performance for each information source in the multimedia system. The fusion method generally improves the performance, and achieves results better than individual information source.
- Second, due to the nature of Up-Fusion method, which tries to maximize diversity, slightly correlation should be achieved.

As a result, it is reasonable to use pseudo labels as true labels for the new data to update the fusion model. Moreover, the correlation among different information sources represents how they co-vary with each other. When the label of an instance changes, the returns of different information sources will change together. Thus, the pseudo labels, even with some errors, should not effect the correlation of the different information sources much. In case that error happens, the methods like [Chen and Ansari, 1998; Atrey and Saddik, 2008] give more confidence to the information sources that have consistent predictions with pseudo label, which in fact is wrong. With the wrong information sources having more confidence, it may result in more errors in the future and almost cannot recover from it. However, in the proposed Up-Fusion method, the diversity between the correct and wrong information sources can still be maintained when errors occur. By maximizing the diversity, due to its good property, the proposed method still has the potential to maintain more confidence in the correct information sources and thus can handle the erroneous pseudo labels.

It is proved in [Blum and Mitchell, 1998] that:

Lemma 5.4.1. *If concept class C is learnable in the classification noise model, then it is also learnable with (α, β) classification noise so long as $\alpha + \beta < 1$.*

Here, (α, β) classification noise is a setting in which true positive examples are incorrectly labeled (independently) with probability α, and true negative examples are incorrectly labeled (independently) with probability β.

With this lemma, we can further prove in a similar way to [Blum and Mitchell, 1998]:

Theorem 5.4.2. *If $(C_1, \ldots, C_N)$ is learnable in the PAC model with classification noise, then $(C_1, \ldots, C_N)$ is learnable in the online fusion model from unlabeled data only, given N initial weakly-useful predictors $h_1(X), \ldots, h_N(X)$.*

Here, a "weakly useful predictor" h of a function f is defined to be a function such that:

1. $Pr_{\mathcal{D}}[h(X) = 1] \geq \epsilon$, and
2. $Pr_{\mathcal{D}}[f(X) = 1 | h(X) = 1] \geq Pr_{\mathcal{D}}[f(X) = 1] + \epsilon$

for some $\epsilon > \frac{1}{poly(N)}$

Proof. Let $f(X)$ be the target concept and $p = Pr_{\mathcal{D}}(f(X) = 1)$ be the probability that a random example from $\mathcal{D}$ is positive. According to the fusion model, we have h which is a fusion function of $h_1(X), \ldots, h_N(X)$. It is reasonable to assume h will be at least as useful as the worst predictor in $h_1(X), \ldots, h_N(X)$. Thus, h is a weakly useful predictor. Let $q = Pr_{\mathcal{D}}(f(X) = 1 | h(X) = 1)$ and let $c = Pr_{\mathcal{D}}(h(X) = 1)$. So,

$$\begin{aligned} Pr_{\mathcal{D}}(h(X) = 1 | f(X) = 1) &= \frac{Pr_{\mathcal{D}}(h(X)=1|f(X)=1) Pr_{\mathcal{D}}(h(X)=1)}{Pr_{\mathcal{D}}(f(X)=1)} \\ &= \frac{qc}{p} \end{aligned} \tag{5.10}$$

and

$$Pr_{\mathcal{D}}(h(X) = 1 | f(X) = 0) = \frac{(1-q)c}{1-p} \tag{5.11}$$

If we use $h(X)$ as a noisy label of X, this is equivalent to (α, β)-classification noise, where $\alpha = 1 - \frac{qc}{p}$ and $\beta = \frac{(1-q)c}{1-p}$. Based on the assumption that h is a weakly useful predictor, we have $c \geq \epsilon$ and $q - p \geq \epsilon$. The sum of the two noise rates satisfies: $\alpha + \beta < 1$ According to the previous lemma, the theorem is proved. According to PAC learning [Valiant, 1984], in the online fusion model, a performance of error less than or equal to γ with at least $1 - \delta$ probability can be learned. Here, $0 < \gamma, \delta < \frac{1}{2}$. □

With the pseudo-labels, our proposed Up-Fusion method solves the update problem when the labels of new data are not available, which were not handled well in previous fusion methods.

5.4.2 Sliding window

Despite the pseudo labels, the return and correlation of different information sources may vary as time goes by due to the context or the nature of data changes. Thus, in the temporal situation, the recent data instances may be more useful in updating the fusion models because these instances are more likely to have same nature or context. A sliding window of the data instances can be used so that only the recent instances will be considered for obtaining the fusion model. Even for the situation where the nature of multimedia data does not change much with the passage of time, the sliding window can also reduce the computation complexity and the memory usage. Thus, it is helpful to have a sliding window on the data for update to cope with the varying situations. To achieve this, at iteration t, the return R_i^t and covariance Φ_{ij}^t are re-calculated using instances X_{α_t} to X_{β_t} as:

$$R_i^t = E[r_{i;X_{\alpha_t:\beta_t}}]$$

$$\Phi_{ij}^t = E[(r_{i;X_{\alpha_t:\beta_t}} - R_i^t)(r_{j;X_{\alpha_t:\beta_t}} - R_j^t)]$$

Here,

- $\beta_t = N_t$, which means X_{β_t} is the newest data instance.
- $\alpha_t = N_t - \pi + 1$, where π is the size of the sliding window. π most recent instances (X_{α_t} to X_{β_t}) are considered for the update of fusion model.

The return and correlation of different information sources may change in different contexts. The window size is generally chosen to be the size of the minimum context duration expected. In this way, the instances in the same context are used to obtain a more proper fusion model. Thus, the proposed Up-Fusion method can deal with changing context or nature of data, which is another advantage of our work.

5.5 Experiments

To show the effectiveness of the proposed Up-Fusion method, experiments have been conducted on both concept detection on TRECVID 2007 dataset [Smeaton, Over, and Kraaij, 2009] and human detection on recorded multiple sensors dataset. The two experiments are representative: concept detection is an important task in information retrieval with average precision as performance measurement, and human detection is a fundamental task in surveillance security with accuracy as performance evaluation. The TRECVID 2007 dataset is one of the mostly used dataset in concept detection task, while the recorded multiple sensors dataset represents the typical multiple sensor surveillance scenario.

The performance is compared with the popular state-of-the-art fusion methods: average fusion method and super-kernel fusion method. The average fusion method simply assigns equal weights to different information sources. It is the most widely used fusion method, and is reported to have good performance for concept detection in [Li et al., 2009]. Super-kernel fusion method [Wu et al., 2004] determines the optimal combination of information sources by further training the output decision scores of different information sources with SVM.

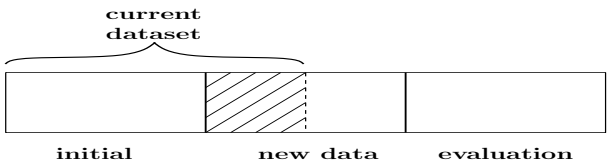


Figure 5.2: The illustration of the experiment setup

5.5.1 Experiment Setup for concept detection

For the concept detection on the TRECVID 2007 dataset [Yanagawa et al., 2007], the models are trained using three visual features: edge direction histogram (EDH), Gabor (GBR), and grid color moment (GCM) [Yanagawa, Hsu, and Chang, 2006]. Table 5.2 shows the description of each feature.

Name of Features	Number of Dimensions
Edge Direction Histogram (EDH)	73
Gabor Texture (GBR)	48
Grid Color Moment (GCM)	225

Table 5.2: Description of the features

There are 21,532 instances in the dataset. The data are evenly divided into three parts: the initial part, the new data part, and the evaluation part. It is illustrated in Figure 5.2. The initial part is taken as the initial seed dataset. The new data part is used to simulate adding new data instances. Then, we evaluate the performance for different concepts on the evaluation part of the dataset.

In the evolution step, at each iteration, we sequentially include $K = 1,000$ instances from new data part into the available dataset and update the fusion models using the proposed Up-Fusion method. The average precision for different concepts is used to evaluate the performance. Here, the average precision for each concept is calculated over the $2,000$ retrieved relevant shots. In this experiment, a total 32 concepts are evaluated, such as Airplane, Animal, Boat_Ship Building, Bus, Car, Charts, *etc.*. The mean average precision (MAP) which is the mean of average precision (AP) for all concepts is used as the performance evaluation criterion.

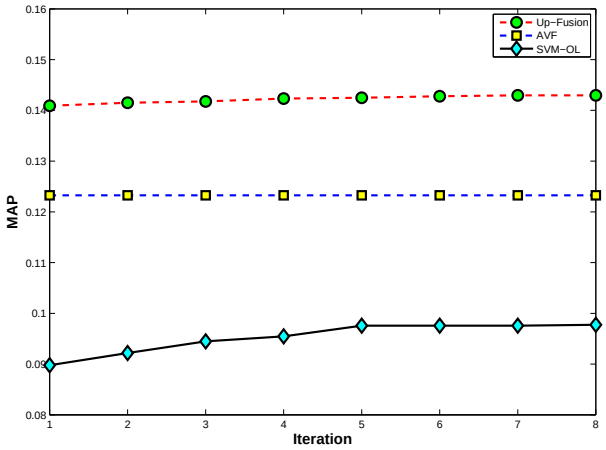


Figure 5.3: MAP based on whole exact return and covariance with true labels. Circle denotes the results of proposed method Up-Fusion, square denotes the results of average fusion method, while diamond denotes the results of SKF-OL

5.5.2 Results

For complete comparison, we give an online version of the super-kernel fusion method by re-training the fusion model with SVM at each iteration. For simplicity, some notations are given here: the average fusion method is denoted as AVF, the super-kernel fusion method is denoted as SKF, the portfolio fusion method is denoted as PTF, the proposed Up-Fusion method is denoted as Up-Fusion, and the online version of super-kernel fusion method is denoted as SKF-OL. Moreover, by default the fusion method means the one based on all the past return and covariance. We add -Win to denote the fusion method with sliding window, and add -P to denote the fusion method with pseudo labels. For SVM training, LIBSVM [Chang and Lin, 2001] is used with RBF kernel and default parameter values. $\lambda = 1$ is used.

The MAP for each iteration based on whole exact return and covariance with true labels is shown in Figure 5.3.

The MAP for each iteration based on return and covariance in sliding window with true labels is shown in Figure 5.4. Here, the window size π is empirically set to 1,000.

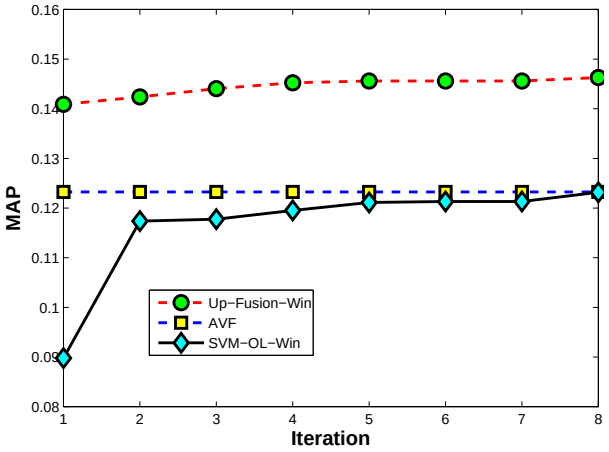


Figure 5.4: MAP based on windowed return and covariance with true labels. Circle denotes the results of proposed method Up-Fusion-Win, square denotes the results of average fusion method, while diamond denotes the results of SKF-OL-Win

The MAP results of different fusion methods are given in Table 5.3. Compared to the MAP of average fusion method, which is 0.123, the final MAP for Up-Fusion method on whole data is 0.143. The final MAP for Up-Fusion-Win method is 0.146. Compared to the portfolio fusion method that utilizes the initial dataset only and stays unchanged as data increase, the proposed Up-Fusion method improves the performance by evolving the fusion models as new data are added. The Up-Fusion method improves PTF by 1.4%(relative), and Up-Fusion-Win method improves by 3.5%(relative). Compared to other fusion methods, the improvement is larger.

We further evaluate the MAP on the situation in which the true labels are unknown and pseudo labels are used as labels. Figure 5.5 shows the MAP for each iteration based on the whole exact return and covariance with pseudo labels. Figure 5.6 shows the MAP for each iteration based on windowed return and covariance with pseudo labels.

The MAP results of different fusion methods are given in Table 5.4. The final MAP for Up-Fusion-P is 0.142. The final MAP for windowed fusion Up-Fusion-Win-P is 0.143. First of all, the proposed Up-Fusion methods on data without true labels

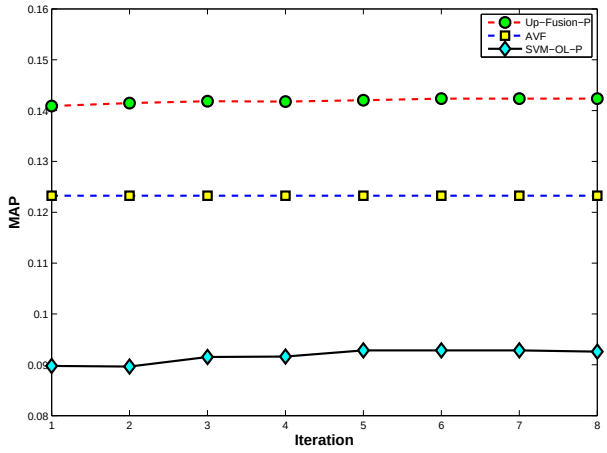


Figure 5.5: MAP based on whole return and covariance without true labels. Circle denotes the results of proposed method Up-Fusion-P, square denotes the results of average fusion method, while diamond denotes the results of SKF-OL-P

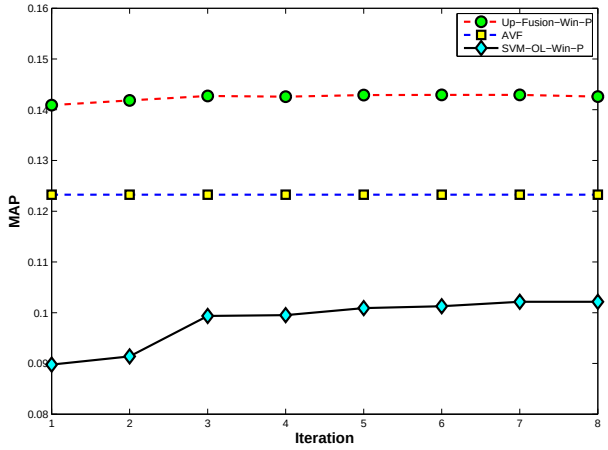


Figure 5.6: MAP based on windowed return and covariance without true labels. Circle denotes the results of proposed method Up-Fusion-Win-P, square denotes the results of average fusion method, while diamond denotes the results of SKF-OL-Win-P

Methods	MAP
AVF	0.123
SKF	0.09
PTF	0.141
SKF-OL	0.098
Up-Fusion	0.143
SKF-OL-Win	0.123
Up-Fusion-Win	0.146

Table 5.3: Performance comparison of different fusion methods on data with true labels

Methods	MAP
SKF-OL-P	0.093
Up-Fusion-P	0.142
SKF-OL-Win-P	0.102
Up-Fusion-Win-P	0.143

Table 5.4: Performance comparison of different fusion methods on data without true labels

outperform the other representative fusion methods. Furthermore, compared to the performance of Up-Fusion methods on data with true labels, the methods with pseudo labels still have comparable performance.

5.5.3 Discussion

Generally speaking, the proposed method obtains better performance than the average fusion method and super-kernel fusion method. The proposed fusion method in the case of unknown labels also demonstrates the superiority over the other fusion methods. The evolution phase generally improves the results. However, the improvement of MAP on concept detection is not quite much. It results from the fact that the distribution and nature of the data in this experiment does not change much, so does the correlation between different information sources. Thus, the update of correlation in each iteration only slightly improves the performance because of more data. Moreover, according to the experimental results, the fusion methods with pseudo labels are comparable to the ones with true labels. Surprisingly, the performance of the online super-kernel fusion method is generally the worst when it takes the new data into account. It may be because the generalization performance tends to suffer when there is too much noise and unbalanced limited data. When training SVM models, it is important to maintain balance between the number of positive and negative instances provided [Yanagawa et al., 2007]. However, given a limited amount of data, maintaining balance is difficult to achieve, especially for the windowed version.

5.5.4 Experimental Setup For Human Detection

To show the utility of our proposed Up-Fusion method, we present the experimental results of human detection (to detect whether there is a human in the region) in the recorded data of a multimedia surveillance system. The experiments are designed with the following two objectives:

- to demonstrate that the proposed Up-Fusion method works well. The overall accuracy of the human detection task, when the proposed method is used, should be better.
- to compare the performance of Up-Fusion with pseudo labels to that with true labels.

The dataset and the experiment setup can be found in Chapter 4 Section 4.5.

The model for each information source is trained using LIBSVM with default parameters. Here, experiments with $K = 1$ and $\pi = 10$ are tested. For the case $K = 1$, we can further compare our proposed fusion method with the confidence evolution method [Atrey and Saddik, 2008] because the method is designed for updating confidence whenever one new data instance is added. The method is denoted as CFE in this chapter.

5.5.5 Results and discussion

The overall accuracy is used as the performance evaluation measure. The experimental results are shown in Table 5.5. As can be seen, the average fusion method gets results of 83.6% accuracy, the confidence evolution method has 82.4% accuracy.

- For the situation that the true labels of the new instances are available, the super kernel fusion method achieves 95.5%, while the proposed Up-Fusion method achieves performance with about 97.4%. The Up-Fusion method outperforms SKF

Methods	Overall Accuracy
AVF	83.6%
CFE	82.4%
SKF-OL-Win	95.5%
Up-Fusion-Win	97.4%
SKF-OL-Win-P	95.5%
Up-Fusion-Win-P	96.7%

Table 5.5: Performance comparison of different fusion methods on human detection

method by 2%(relative). Compared to the AVF method, the Up-Fusion method improves the performance by 16.5%(relative). Compared to the CFE method, the Up-Fusion method improves the performance by 18.2%(relative).

- For the situation that the true labels of the new instances are not available, the super kernel fusion method achieves 95.5%, while the proposed Up-Fusion method achieves performance with about 96.7%. The Up-Fusion method outperforms SKF method by 1.3%(relative). Compared to the AVF method, the Up-Fusion method improves the performance by 15.7%(relative). Compared to the CFE method, the Up-Fusion method improves the performance by 17.4%(relative).

Generally speaking, the proposed Up-Fusion method outperforms the other methods. Consider the situation that the human is occluded, if the video has been considered reliable before, the fusion decision may be wrong. The methods like CFE will give more confidence to the video and thus may not recover from the error. However, with the Up-Fusion method, the diversity between video and audio will still be increased due to different predictions. Thus, the audio can still be given a high weight and have the chance to recover from the error. This might explain why Up-Fusion outperforms the other methods. Moreover, it can be seen that the results of the fusion methods with pseudo labels are comparable to the ones with true labels.

5.6 Conclusions

In this chapter, an evolving fusion method has been proposed. Compared to the previous static fusion methods, especially the portfolio fusion method, as new data are continually added, the proposed Up-Fusion method evolves to adapt to the changing data and environment conditions. Evolved fusion models for different conditions can perform better than a fixed fusion model. Compared to the previous evolving fusion methods, our method utilizes the correlation among different information sources, can deal with either crisp or soft decision, and no context information is required. Pseudo labels are used in the case when the label information of newly added data is not available. A sliding window has been introduced to deal with the temporal change of multimedia data. Experiments on representative concept detection and human detection tasks have shown the superiority of the proposed Up-Fusion method. Better updating methods will be studied in the future. The fusion performance in the situation where the context information is available will also be investigated.

Chapter 6

Specialist Fusion

In multimedia fusion, different information sources generally do not have consistent performance on different data instances. Some may predict correctly on one instance, but cannot perform well on another data instance. Particularly, we say the information source is an expert for a data instance if it predicts correctly on the instance. If we can predict whether an information source is an expert for a data instance, the information sources can then be combined using appropriate fusion model based on the prediction. This way should improve the fusion performance. As a result, a specialist fusion method is proposed in this chapter. The intrinsic connection to our daily life experiences provides a very strong psychological basis. Our goal in considering the decisions of multiple experts, is to improve our confidence that we are making the right decision. Seeking opinions from specialists before making a decision is an innate behavior for most of us. Given multiple experts, they may be experts in different areas: one may be expert in one area but not expert in another area. When we have some problem to consult, instead of seeking opinions from experts in all different areas, we usually consult the relevant experts in the particular area of the problem. For example, when we have a software problem, we may would like to consult a software engineer instead of a pharmacist. For the problems in different areas, we may consult different experts.

The rest of this chapter is organized as follows: Section 6.1 reviews related work. Section 6.2 describes the architecture and details of the proposed specialist fusion method. Experiments setup and results are shown and discussed in Section 6.3. In the end, Section 6.4 gives the conclusion.

6.1 Related Work

For multimedia decision fusion, the decisions from individual information sources are first obtained. Then, different strategies are applied on the individual decisions to combine them for the final decision. Among these strategies, Linear Opinion Pool is one of the most widely used methods. This method attaches a measure of value such as a weight to the decision provided by each information source. In this way, the decisions from different information sources are linearly combined. Usually equal weights are set to different information sources. This way is referred to as *average fusion*. Recently, a fusion method that sets weights using portfolio theory has been proposed in [Wang and Kankanhalli, 2010b]. The weights for different information sources are obtained by maximizing expected return while minimizing risk to achieve an overall good performance. It is referred to as *portfolio fusion*. The linear fusion method is limited by the linear-model complexity [Wu et al., 2004]. Then, a more sophisticated *super-kernel fusion* method has been proposed in [Wu et al., 2004], which utilizes high model complexity to explore interdependencies between information sources. The decision scores are concatenated into vectors, and again, SVM is employed to yield the fusion model with the decision vectors as input.

In general, the previous fusion methods, both the linear and non-linear fusion models, tend to have one single static fusion model on all the data. However, in reality, even for the same task, different information sources generally do not have consistent performance in different situations. The information source may work well in some

situation while does not perform well in another situation. For example, in surveillance systems, the visual camera may perform better in detecting person in normal situation, but the audio sensor may be better when there are occlusions. Thus, one single fusion model may not work well on all the data.

Some context aware fusion methods have been proposed like [Lee and Park, 2008; Geng et al., 2010]. In *context weight fusion method* [Lee and Park, 2008], adaptive weighting scheme was adopted for acoustic and visual speech recognition. The weights for audio and visual vary according to the noise level in speech. These context aware fusion methods have different fusion models according to different contexts. Based on the context information, the appropriate fusion model is chosen and better performance is achieved. However, the methods need the context information which may not be available and dealing with all influential context factors is unrealistic.

In the specialist fusion method, an expert predictor of the information source is introduced. For any data instance, the decisions from different information sources are combined based on their expert predictions. In this way, the decisions of expert information sources are fused as the final decision, which should be better than combining the decisions from all the information sources without distinction. Moreover, the expert prediction is based on the data instance and no context information is required.

There are also some similar works in meta-learning literature, especially in the area of building meta-rules matching task properties with algorithm performance [Vilalta and Drissi, 2002], such as landmarking [Bensusan and Giraud-Carrier, 2000; Pfahringer, Bensusan, and Giraud-Carrier, 2000]. The idea is to choose that learning algorithm displaying best performance around the neighborhood of the test example [Keller, Paterson, and Berrer, 2000; Brazdil and Soares, 2000]. By gathering the k-nearest neighbor examples of a test example in the meta-domain, the method simply selects the learning algorithm with best averaged performance around the neighborhood of the test example. The method is thus referred to as best neighborhood selection

method (BNS) in the remaining part. It is worthy to mention that the best neighborhood selection meta-learning algorithm is to select the different learning algorithms that are trained on the homogeneous feature set. It is different from fusion method that combines information in different modalities. Moreover, the BNS method tries to predict the best algorithm using KNN method and use the algorithm to predict the test example. First, the method uses only one information source. Second, the model for predicting the best algorithm is also quite simple. In our specialist fusion method, we use more sophisticated classification model to predict the expertise of information source. Then, the decisions together with the expertise from all the information sources are combined in a linear fusion, which utilizes all the information sources and is more tolerant to noise.

Collaborative filtering technique is also a popular way of fusing opinions from different users for recommendation. It assumes that "users who have similar preferences in the past are likely to have similar preferences in the future, and the more similar they are, the more likely they would agree with each other in the future" [Jambor and Wang, 2010]. In multimedia classification application, we may need to consider each example as a user. For a test example, the most similar example can be selected and used for the prediction of the test example. In this way, the best neighborhood selection method can be considered as a collaborative filtering technique in our multimedia classification applications.

6.2 Proposed Method

For a multimedia data analysis task, there are N information sources $M_1, \ldots, M_N$. The proposed specialist fusion architecture is depicted in Figure 6.1.

The algorithm of specialist fusion consists of the following steps:

1. *Training classification models*: Given the training data, obtain the classifica-

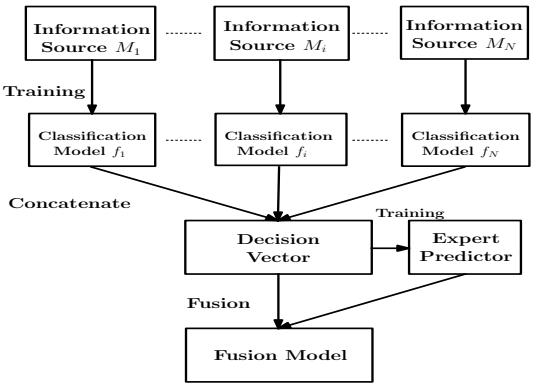


Figure 6.1: The proposed fusion architecture

tion model for individual information source. For each data instance $\mathbf{X} = \{\mathbf{X}^{(1)},\mathbf{X}^{(2)},\ldots,\mathbf{X}^{(N)}\}$, the corresponding label $y(\mathbf{X})$ is available in the training data. Here, $\mathbf{X}^{(i)}$ is the observation from information source M_i. Then, for information source M_i, the training instances $\mathbf{X}^{(i)}$ and their corresponding labels can be obtained. The classification model f_i can then be trained through learning algorithm. Though many learning algorithms can be employed, SVM is employed here because of its effectiveness.

2. *Training expert prediction models*: Obtain the expert predictors for individual information source. With the N classification models from different information sources, N decision scores can be obtained for each data instances. The decision scores are then composed into a decision vector $V(\mathbf{X}) = [f_1(\mathbf{X}),\ \ldots,\ f_N(\mathbf{X})]$. Moreover, for information source M_i, the expert label for each data instance $\mathbf{X}$ can also be obtained based on the classification performance of f_i. It is worthwhile to mention that the decision score s from SVM is in $[0, 1]$. The predicted class membership is based on $sign(s - 0.5)$. The expert label e can then be defined as:

$$e(\mathbf{X}) = \begin{cases} 1 & \text{if } sign(f_i(\mathbf{X}) - 0.5) = y(\mathbf{X}) \\ -1 & \text{Otherwise} \end{cases} \tag{6.1}$$

That is, if one information source predicts correctly on $\mathbf{X}$, its expert label on $\mathbf{X}$ is

+1. Otherwise, it is −1. The decision vectors and the corresponding expert labels (V, e) are then treated as training instances. The expert predictor for individual information source can be trained using SVM. The expert predictor for information source M_i is denoted as D_i. Taking decision vector $V(\mathbf{X})$ as the input to the model D_i, the prediction $D_i(\mathbf{X})$ can be considered as the confidence that M_i is an expert on instance $\mathbf{X}$.

3. *Test situation*: For the test data instance, fuse the decisions from different information sources to obtain the final decision. For data instance $\mathbf{X}$, the decision vector $V(\mathbf{X}) = [f_1(\mathbf{X}), \ldots, f_N(\mathbf{X})]$ can be obtained according to the models $f_i, i = 1, \ldots, N$ obtained in step 1. With the decision vector as input, the expert predictions $P(\mathbf{X}) = [D_1(\mathbf{X}), \ldots, D_N(\mathbf{X})]$ can be obtained based on the expert prediction models $D_i, i = 1, \ldots, N$ obtained in step 2. Then, the decisions from different information sources are combined to obtain the final decision as:

$$I(\mathbf{X}) = sign((V(\mathbf{X}) - 0.5)(P(\mathbf{X}) - 0.5)^T) \tag{6.2}$$

$I(\mathbf{X})$ is the class membership for $\mathbf{X}$. That is, the decisions from classification models are weighted combined using their corresponding expert confidences.

The fusion procedure is illustrated in Figure 6.2.

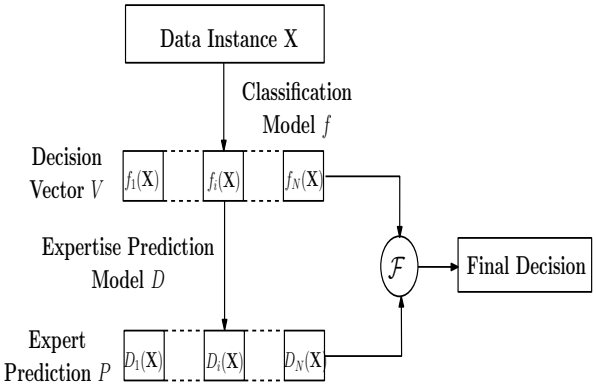


Figure 6.2: Specialist Fusion Method

6.3 Experiments

Our experiments were designed to evaluate the effectiveness of the specialist fusion method. Specifically, we want to compare its performance with three popular state-of-the-art fusion methods: super-kernel fusion method (SKF), average fusion method (AVF), and portfolio fusion method (PTF). We also compared with the best neighborhood selection method (BNS). Here, we measured the neighborhood of a test example in the space of meta-features (decision vector), and simply set $k = 1$ for k-nearest neighbor [Bensusan and Giraud-Carrier, 2000]. We conducted our experiments on image aesthetics inference and affective image classification problems. They are appropriate problems to test our method: Judging aesthetic qualities or emotional content of photographs is a highly subjective task. It is also very useful to measure the aesthetic qualities or emotions for image retrieval. More importantly, it is still unclear what properties may have correlation with aesthetics or emotions and how they are important to aesthetics or emotions. Generally speaking, many kinds of features from different information sources will be used for both problems. Thus, a proper fusion method to combine the information sources is important for improving performance.

Two real-world datasets are used in this experiment: one is an image aesthetics dataset [Datta, Li, and Wang, 2008], and the other is an affective image dataset [Machajdik and Hanbury, 2010]. The image aesthetics dataset contains 3,581 images downloaded from Photo.net. The task is to distinguish between photographs of high and low aesthetic values. According to the analysis in [Datta et al., 2006], two classes of data are chosen, high containing samples with aesthetics scores greater than 5.8, and low with scores less than 4.2. The affective image dataset is a set of 806 artistic photographs downloaded from deviantart.com. The task is to classify images into emotional categories: Amusement, Awe, Contentment, Excitement as positive emotions, and Anger, Disgust, Fear, Sad to represent negative emotions [Machajdik and Hanbury, 2010]. The

Category	Number
Amusement	101
Awe	102
Contentment	70
Excitement	105
Anger	77
Disgust	70
Fear	115
Sad	166

Table 6.1: Number of images per emotional category in affective image dataset

details for affective image dataset are shown in Table 6.1.

- For the image aesthetics inference task, the dataset is evenly divided into training and testing parts, with two parts containing the same number of high and low aesthetics images. A total 59 features in [Datta, Li, and Wang, 2008] are extracted for aesthetics inference, such as Brightness, Contrast, Image aspect-ratio, Wavelet feature, *etc.*.

- For the affective classification task, it is considered as several binary classification problems: for each category, to classify whether the images belong to this category. For each category, the dataset is evenly divided into training and testing parts. A total 10 features in [Machajdik and Hanbury, 2010] are employed. They are Saturation, Brightness, Pleasure-Arousal-Dominance, Hue, Colorfulness, Tamura, Wavelet textures, GLCM features, Low Depth of Field (DOF), and Rule of Thirds. Each category is separated against all others and trained using one-against-all method.

The classification models are trained for individual information sources by applying SVM. For all the SVM training procedures, LIBSVM [Chang and Lin, 2001] is used with RBF kernel and default parameters. In general, the instances in the dataset are highly skewed towards negative samples. Thus, the instances are sub-sampled for balanced data in a similar way to [Yanagawa et al., 2007]. The procedure of sub-sampling is

Method	SKF	AVF	PTF	BNS	SPF
Accuracy	62.4%	53.4%	60.0%	60.0%	**63.0%**

Table 6.2: Comparison of different methods on image aesthetics inference performance

as follows: Let N_p denote the number of positive instances, and N_n denote the number of negative instances in the dataset. If $N_p < N_n$, we randomly selected a set of negative instances equal in size to the number of positive instances.

The proposed specialist fusion method is compared with SKF, AVF, PTF, and BNS on both tasks. For each experiment setup, the average classification accuracy over 10 runs is taken as the result to reduce the effects of randomness. The classification accuracy of image aesthetics inference for individual information source is shown in Figure 6.3. The performances of proposed specialist fusion method (SPF) and other methods are given in Table 6.2.

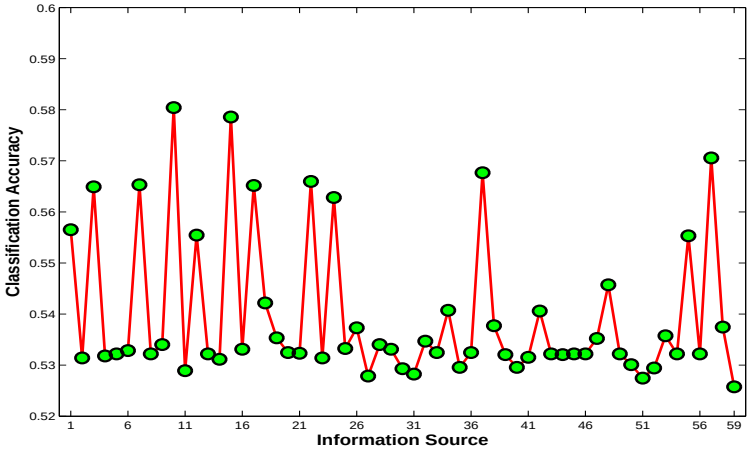


Figure 6.3: The performance of image aesthetics inference for individual information source

As it can be seen, in image aesthetics inference task, the proposed SPF method outperforms the other methods. The SPF method improves the performance of SKF by around 1%(relative), and outperforms the best information source(M_{10}), whose accuracy is 58%, by 8.6%(relative). The correlation of different information sources for image aesthetics inference is shown in Figure 6.4. It shows the information sources generally

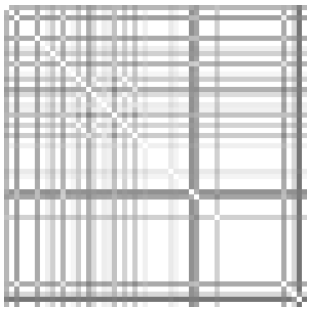


Figure 6.4: Correlation of different information sources for image aesthetics inference

have consistent performance. The improvement over SKF is not much may be because the information source set contains many noisy information sources with low accuracy and no single information source that performs very well (the accuracies of all individual information sources are around 52%–58%). We also compared SPF with feature fusion (FF), which uses a single SVM taking all features. The performance of FF (67%) is slightly better than SPF because the performance of individual feature is similar and thus the SPF does not improve the performance much. Another reason for feature fusion performs better may be that some information may be lost in decision.

The performance of affective classification for individual information source is shown in Figure 6.5.

The performances of proposed specialist fusion method (SPF) and other methods are given in Table 6.3.

Method	**SKF**	**AVF**	**PTF**	**BNS**	**SPF3**
Amusement	58.5%	85.9%	86.3%	76.3%	**87.1%**
Anger	37.5%	9.5%	51.7%	73.9%	**89.7%**
Awe	35.5%	21.7%	57.4%	69.2%	**86.2%**
Contentment	41.2%	29.2%	54.7%	71.6%	**91.1%**
Disgust	35.8%	21.2%	55.2%	73.6%	**90.8%**
Excitement	49.2%	21.5%	62.4%	76.1%	**86.5%**
Fear	51.7%	14.2%	78.9%	78.0%	**84.6%**
Sad	49.0%	20.6%	43.4%	68.0%	**76.1%**
Average	44.8%	28.0%	61.3%	73.3%	**86.5%**

Table 6.3: Comparison of different methods on affective image classification performance

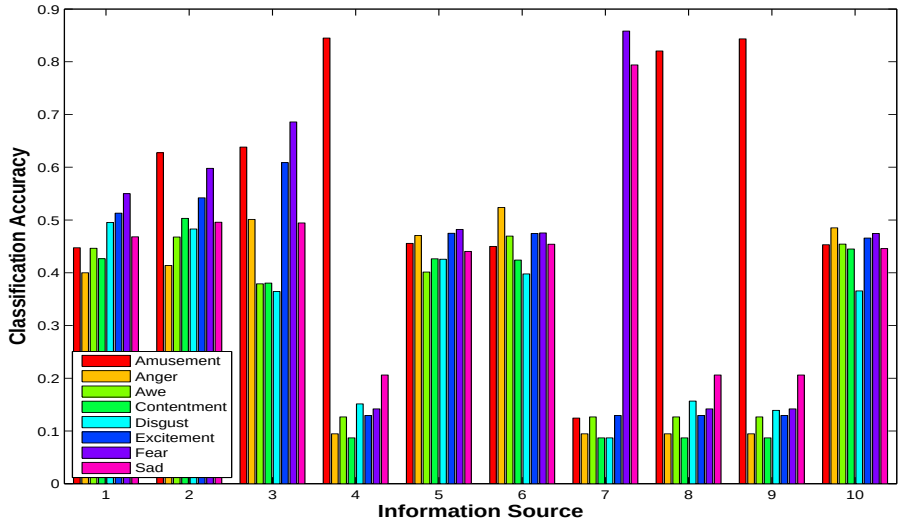
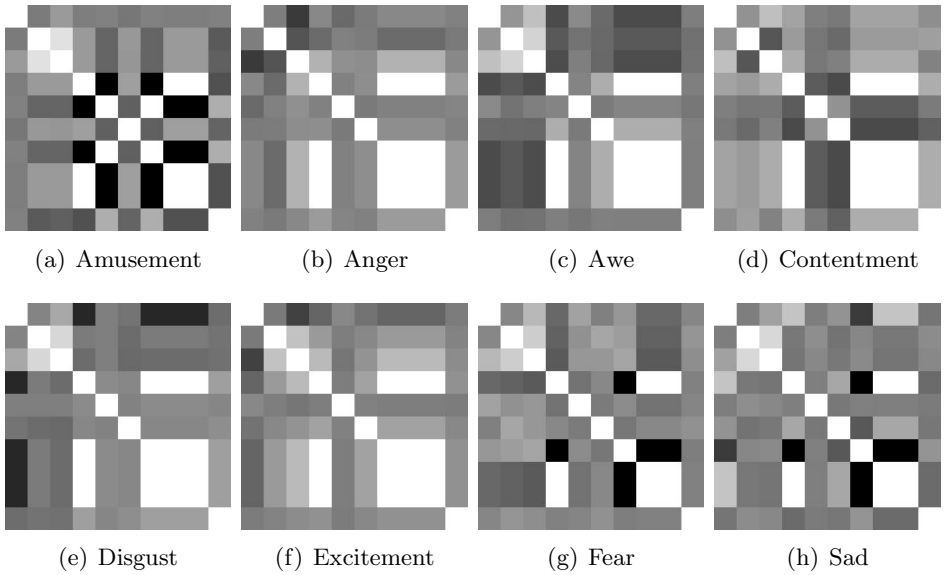


Figure 6.5: The performance of affective classification for individual information source

It can be seen that the SPF generally outperforms the other methods in affective image classification for different emotional categories. On average, SPF improves the performance of PTF by about 41.1%(relative). SPF also outperforms the best information source(M_2), which achieves 51.6% accuracy. In fact, for the affective image classification problem, some of the information sources such as M_8 have an accuracy that is as low as 22% on average. This is also why AVF generally has poor performance. Moreover, the SPF significantly outperforms the FF (21.6%).

The correlation of different information sources for affective image classification is shown in Figure 6.6. It can be seen that the information sources in image affective are less correlated than those in image aesthetics. It is consistent with the performance in these two datasets and explains why the improvement on affective image data is larger.

Significance tests have also been conducted on the proposed method. The accuracies of the 10 runs on the different tasks of image aesthetics inference and affective image classification are obtained. The proposed SPF passes the *t-test*s with SKF, AVF, PTF, and BNS at the 5% significance level. Moreover, the permutation test have also been done. For $n = 1,000$, the $p - value$ between SPF and all other fusion methods are



(a) Amusement (b) Anger (c) Awe (d) Contentment

(e) Disgust (f) Excitement (g) Fear (h) Sad

Figure 6.6: Correlation of different information sources for affective image classification

0.

6.4 Conclusions

Based on the common practice of seeking opinions from specialists before making a decision, a specialist fusion method is proposed in this chapter. The notion of an expert predictor for individual information source on different situations is introduced. With the expert predictions, the decisions from different information sources are combined to obtain the final results. Experiments have been conducted on both image aesthetics inference and affective image classification tasks. The experimental results show the superiority of the proposed specialist fusion method compared to other popular fusion methods. When there are too many noisy information sources, the improvement is not much. Thus, a good feature selection method before fusion or better combination method will be investigated in the future. Moreover, the possibility of applying collaborative filtering techniques to multimedia fusion will also be studied.

6.5 Further Comparison

Till now, four multimedia decision fusion methods have been presented in this dissertation: MultiFusion method (MF), portfolio fusion method (PTF), Up-Fusion method, and specialist fusion method (SPF). Among the four methods, Up-Fusion is used to evolve the fusion model using correlation, and can be used in the system with increasing amount of data. For the other three methods, they can be used to classification problem. Thus, we further compare the three methods in different scenarios with simulation data. We simply test the performance of the three fusion methods in 6 scenarios, which are listed in Table 6.4. In each scenario, we simulate 10 information sources $M_1 - M_{10}$ and fuse them. For balanced data, we generate 100 positive and 100 negative training instances. For unbalanced data, we generate 20 positive and 200 negative training instances.

Scenario	Data	Correlation	$\mu_1 - \mu_0$	σ
1	balanced	Independent	10	i for M_i
2	unbalanced			
3	balanced	Correlated ($M_1 - M_9$)	10	9 for $M_1 - M_9$, 1 for M_{10}
4	unbalanced			
5	balanced	Correlated ($M_2 - M_{10}$)	1	1
6	unbalanced			

Table 6.4: Description of simulation scenarios

The simulation results are shown in Table 6.5. It can be seen that all the methods can perform well when the information sources are independent and have different performances (scenarios 1 and 2). Among the different scenarios, MF can perform well when there are unbalanced noisy data (*e.g.*, scenario 4). PTF generally shows good performance, especially when there are correlated information sources (*e.g.*, scenario 3). But it does not give performance as good as MF when the data are unbalanced (*e.g.*, scenario 4). SPF generally works well and can handle the unbalanced noisy data. It performs the best on independent data. But it may not perform well when the information

sources give consistent performance (*e.g.*, scenario 3 and 6).

Scenario	MF	PTF	SPF
1	100%	100%	100%
2	98.7%	99.6%	99.8%
3	99.9%	100%	94%
4	99.8%	91.4%	99.6%
5	67.8%	70.9%	68.8%
6	90.1%	90.7%	87.1%

Table 6.5: Performance of different fusion methods in simulation

In summary, all the proposed fusion methods can fuse the information sources to improve the performance. PTF works well when the information sources are diverse and some are positively correlated. SPF works well when the different information sources do not have consistent performance on data instances. MF can perform well on unbalanced noisy data since it uses the boosting structure. However, there are also some limitations. For example, MF uses weighted majority voting and thus can be dominated by poor information sources that in majority. PTF has sophisticated correlation model and improve the performance by minimizing correlation and maximizing expected performance. But it is still linear fusion and data independent fusion method. SPF is a promising attempt to adopt data dependent fusion model with linear weighted sum. But it cannot improve the performance much if the information sources have similar performance. More thorough study will be investigated. Up-Fusion is an attempt to evolve fusion model with increasing data. Better updating methods will be studied in the future.

Chapter 7

Conclusions

In this chapter, we summarize the conclusions that we have reached in multimedia decision fusion. Also, a few potential areas for extension and possible applications of these research results will be presented.

7.1 Summary of Research

This dissertation proposes to develop more comprehensive methods for multimedia decision fusion. The aim is to properly utilize the correlation among multimedia information sources to achieve goals such as concept detection, and surveillance. We argue that multiple information is useful in drawing more accurate inferences and in making better decisions. Use of multimedia fusion raises several research issues such as how to utilize the correlation among different multimedia information sources, how to cope with the uncertainty and diversification of multimedia information, and how to adapt to the changing and increasing data. The dissertation has proposed several multimedia decision fusion methods to address these issues. Below, we summarize the specific contributions and findings of our works.

7.1.1 MultiFusion Method

We began by developing a multimedia decision fusion method to make better use of the correlation among multimedia information sources. It was observed that the correlation was usually used only once in the multimedia fusion task. Therefore, in Chapter 3, we have presented a novel *MultiFusion* approach to make better use of multimedia correlations. By adopting a boosting structure, the correlation of multiple information sources is used to form a multimodal classifier in each iteration. Therefore, the multimedia correlation is used multiple times instead of being combined once. How to apply *MultiFusion* to multimodal data in different applications to utilize complementary yet correlated information and improve the performance is shown in the chapter. A framework for adopting Adaboost-like structure to utilize correlation multiple times is described. The fusion model for each iteration and for all the intermediate classifiers are also presented in this chapter. By comparing with other boosting structure fusion methods, we have demonstrated in both simulation experiments and the real application task that multiple utilization of correlations among different information sources helps in obtaining more accurate and credible decisions. The fusion performance obtained by MultiFusion method outperforms several other related fusion methods, such as Learn++ Fusion [Parikh et al., 2004] and Selection Fusion [Xue and Ding, 2006]. The results in human detection corresponds well with the observation obtained in simulation experiments. The experiment results suggest that the novel use of multiple utilization of correlations can improve the fusion performance.

7.1.2 Portfolio Fusion Method

It was also observed that there is always an uncertainty problem in multimedia information fusion and diversity in information sources can improve the performance. Classification based on any one information source is usually not perfect. Thus, the

decision cannot be estimated with absolute certainty using the classification models. There are many sources of uncertainty such as ambiguity, noise, and deviations between the scoring function and the true probability of relevance. Uncertainty is an extremely important feature that demands serious consideration. Even when we have a classifier with high expected accuracy, it is not safe if its variance is high [Breiman, 1996]. Moreover, the information sources in the multimedia systems are generally correlated. Thus, diversification is beneficial for multimedia fusion. Both uncertainty and correlation should be considered in multimedia fusion. Therefore, in Chapter 4, we have proposed the portfolio fusion method to maximize the return and minimize the risk (uncertainty) by introducing the widely used and effective portfolio theory from finance. A more sophisticated technique to utilize correlations among different information sources is presented. The method using portfolio theory that attempts to maximize expected performance and minimize risk to achieve a high dependable performance is described in this chapter. Each information source in the multimedia system is considered the equivalent of a security in financial investment. By properly modeling the return and risk of information sources, portfolio theory gives the optimal weights for the information sources by minimizing the risk while maximizing the return. The experimental results in simulation and concept/human detection have demonstrated that the portfolio fusion makes better use of correlations and helps in achieving a better performance. Based on the results, it appears that the method is able to utilize the correlation and model uncertainty properly. It is shown that the method is also easily scalable because our proposed fusion method does not require additional learning for weights after models for each modality are trained. This may be attributed to the proper modeling of correlation and uncertainty. To the best of our knowledge, this study is the first to explicitly consider both the uncertainty and correlation issues in multimedia decision fusion.

7.1.3 Up-Fusion Method

For multimedia analysis, it is common that little amount of training data are available at the beginning. Furthermore, the multimedia data keep increasing with time and the nature of the data collection can change. Most of the traditional fusion methods are static with respect to time. To address this, in recent years, several evolving fusion methods have been proposed. However, they can only be used in limited scenarios. Thus, Chapter 5 has described an attempt to evolve the multimedia fusion model and improve the performance with new data to deal with such situations. A novel evolving fusion method, called *Up-Fusion*, is proposed based on the online portfolio selection theory in this chapter. Evolved fusion models for different conditions can perform better than a fixed fusion model. Compared to the previous evolving fusion methods, the proposed method takes the correlation among different information sources into account, and evolves the fusion model when new multimedia data are added. It can deal with either crisp or soft decisions without requiring additional context information. Pseudo-labels are used in the case when the label information of newly added data are not available. A sliding window approach has been utilized to deal with temporal change of the multimedia data. Experiments on representative concept detection and human detection tasks have shown the superiority of the proposed Up-Fusion method. With the utilization of multimedia correlation and refinement, the method evolves the fusion model along with the newly added multimedia data to improve the performance. Moreover, the situations that the labels of newly added data are not available and that context or nature of data changes, are also handled.

7.1.4 Specialist Fusion Method

Many multimedia fusion methods have been proposed to utilize information from different sources to improve the analysis performance. However, how to fuse the information

sources most appropriately have not yet been solved. Seeking opinions from specialists before making a decision is an innate behavior for most of us. Getting back to multimedia fusion, evaluating expertise of information sources before combining the decisions from different information sources should also help to make a better decision. Thus, in Chapter 6, based on the common practice of seeking opinions from specialists before making a decision, a specialist fusion method that adaptively fuses different information sources according to the expertise of different information sources on different data instances is proposed. In multimedia fusion, different information sources generally do not have consistent performance on different data instances. Some may predict correctly on one instance, but cannot perform well on another data instance. Particularly, we say the information source is an expert for a data instance if it predicts correctly on the instance. In this chapter, the notion of an expert predictor for the information source is introduced. For any data instance, the decisions from different information sources are weighted and combined based on their expert predictions. In this way, the decisions of expert information sources are fused as the final decision. The experimental results on both image aesthetics inference and affective image classification tasks have demonstrated the superiority of the proposed specialist fusion method when compared to other popular fusion methods. Particularly, combining the decisions from expert information sources is better than combining the decisions from all the information sources without distinguishing their expertise. Moreover, the expert prediction is based on the data instance and no context information is required.

7.2 Conclusions

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

- The decision level multimedia fusion is advantageous over data/feature level fusion. It offers flexibility and scalability in terms of information sources used in

multimedia analysis task. Data from different information sources can be analyzed using different yet appropriate methods in decision level fusion. It is consistent with *Principle of Least Commitment* and *Principle of Graceful Degradation*. When new information source is introduced, only the model for the new information source as well as fusion model need to be updated. Furthermore, decision level fusion is intuitive and easy to perform, and it is easy to control the relative contributions of information sources to fusion results.

- Use of correlations among the different information sources helps in improving the performance of multimedia analysis task such as concept detection and human detection. MultiFusion method shows that using correlations multiple times can help in improving the performance. Portfolio fusion method further shows that sophisticated modeling of correlations can also improve the performance.

- Utilization of the correlation and uncertainty can help in obtaining a better dependable performance. Portfolio fusion method provides a possible way to combine both correlation and uncertainty.

- Evolved fusion models for different conditions can perform better than a fixed fusion model. With the utilization of multimedia correlation and refinement such as pseudo-labels and sliding window, the fusion model can be evolved along with the newly added multimedia data to improve the performance. Moreover, the situations that the labels of newly added data are not available and that context or nature of data changes, can also be handled.

- Based on the common practice of seeking opinions from specialists before making a decision, combining the decisions from expert information sources can help in obtaining better classification accuracy. The specialist fusion method is a useful attempt and shows promising results.

- Several decision level multimedia fusion approaches have been proposed. They aim to solve different objectives. Thus, the suitable occasions for different fusion methods are as follows: Generally speaking, the MultiFusion method is based on Adaboost-like structure and improves the fusion performance by implicitly using correlation multiple times. It generally works for classification problem. Because the MultiFusion method needs to train $N \times T$ (N is the number of information sources, and T is the number of iterations) classifies in total. The method can be used when there is weak learner for training data and the computational complexity is not critical. Moreover, the method implicitly utilizes the correlation by fusing information sources using weighted majority voting. It may not work as well as portfolio fusion in the case that the information sources are highly correlated. Portfolio fusion method uses a more sophisticated model of correlation. The improvement will be obvious when there are highly correlated information sources in the multimedia system. In this case, the positive correlation may hurt the performance. By minimizing the risk while maximizing the expected performance, portfolio fusion method can achieve diversity in information sources and improve the performance. Both of the above methods are used in the static situation. In the case that the multimedia data keep increasing and the nature of data can change, the static fusion model may not be effective as the data increase. In this case, the Up-Fusion method can be used to evolve the fusion model with newly added data. Specialist fusion method adaptively weights different information sources according to their expert predictions. It should be helpful in improving fusion performance when the information sources do not have consistent performance on all the data, *e.g.*, the data from different contexts. In this case, different information sources may have different performances on data from different contexts. Then, assigning the same confidence on the information source over all the data is not effective. The specialist fusion method can be used to pre-

Method	Strategy	Suitable Occasion
MultiFusion	multiple use of correlation in Adaboost-like structure	classification problem can be weakly learned and high computational complexity is not critical
Portfolio Fusion	portfolio theory with a more sophisticated modeling of correlation	multimedia system with highly correlated information sources
Up-Fusion	evolve the fusion model with newly added data	multimedia system with increasing data, the nature of which can change
Specialist Fusion	adaptively weight information sources according to their expert predictions on different data	multimedia information sources that do not have consistent performance on all the data, *e.g.*, the data from different contexts

Table 7.1: Summary of proposed fusion methods

dict the expertise of the information source on different data instances and gives different appropriate confidences on different data instances for the information source, which can improve the fusion performance. The discussion is summarized in Table 7.1.

7.3 Future Directions

There are some limitations and potential extensions in the areas of research presented in this thesis.

7.3.1 Correlation and Risk Modeling

We have proposed the portfolio fusion method by modeling variance as the risk and mean as the expectation. However, there were some limitations in this model. In our portfolio fusion method, variance as an indicator of the risk does not distinguish a negative return from a positive return. It is thus worthwhile investigating "down-side risk" in finance that considers only bad surprises. For the changing and increasing multimedia data situation, better updating methods can be studied for Up-Fusion in

the future. How to properly model the correlation and update it is also an important issue.

7.3.2 Active Fusion

Current multimedia analysis systems are usually designed to handle the specified task using multimedia fusion methods with the data obtained by the different information sources, such as different heterogeneous sensors. There are few interactions between the fusion module and the information sources. The information sources are designed to obtain the data from environment. Then, the fusion module tries to accomplish the task based on the data that different information sources give. The fusion method can only build fusion model and probe the decision using the input data. Thus, almost all the previous multimedia fusion works are passive fusion. That is, the data are all what the fusion methods have and there is no interaction between the information sources and the fusion module. The fusion methods passively combine what they can get from the data to obtain an improved result compared to using single information source. However, it is worthwhile to extend the fusion to active fusion. The fusion methods not only analyze the obtained data, but also can actively get new useful data based on the analysis. This way should be more useful in many applications. For example, in modern multimedia surveillance systems, the sensors usually have certain freedom of motion or even are completely mobile. Initially, the different sensors observe the environment and send the data to fusion module. The fusion method then constructs the fusion model based on the data and tries to reach a decision. If the data are not enough, the fusion module may further instruct certain sensors to move and probe more data. The interactions can be made until a credible decision has been reached. In addition, the fusion module can even pick proper fusion models or suggest new sensors to accomplish the task.

7.3.3 Context Modeling

Context modeling requires a model that describes the context in a generic and scalable manner. In the multimedia analysis systems, the context of the data usually varies greatly. The performance of different information sources also varies greatly under different contexts. There have been some context-aware fusion methods proposed, such as [Geng et al., 2010]. These methods dynamically determine the relative reliability of different information sources based on the context. The context is usually captured from the environment. However, the context information in some multimedia systems may not be available and dealing with all influential context factors is unrealistic in practice. In order to achieve better fusion performance, the system should be able to identify the context of different data with proper context modeling from the data. Then, the system can choose proper fusion models for different contexts. In this way, better fusion results should be obtained.

References

Adams, Bill, Arnon Amiry, Chitra Doraiz, Sugata Ghosalx, Giridharan Iyengar, Alejandro Jaimesz, Christian Langz, Ching yung Linz, Apostol Natsevz, Milind Naphadez, Chalapathy Neti, Harriet J. Nock, Haim H. Permutery, Raghavendra Singhx, John R. Smithz, Savitha Srinivasany, Belle L. Tsengz, Ashwin T. V., and Dongqing Zhang. 2002. Ibm research trec-2002 video retrieval system. *NIST TREC*.

Adams, WH, G. Iyengar, C.Y. Lin, M.R. Naphade, C. Neti, H.J. Nock, and J.R. Smith. 2003. Semantic indexing of multimedia content using visual, audio, and text cues. *EURASIP Journal on Applied Signal Processing*, 2:170–185.

Alexandre, Luís A., Aurélio C. Campilho, and Mohamed Kamel. 2001. On combining classifiers using sum and product rules. *Pattern Recognition Letters*, 22(12):1283–1289.

Aly, Robin and Djoerd Hiemstra. 2009. Concept detectors: how good is good enough? In *ACM International Conference on Multimedia*, pages 233–242.

Amores, Jaume, Nicu Sebe, Petia Radeva, Theo Gevers, and Arnold W. M. Smeulders. 2004. Boosting contextual information in content-based image retrieval. In *ACM SIGMM International Workshop on Multimedia Information Retrieval*, pages 31–38.

Atrey, P. K. and A. El Saddik. 2008. Confidence evolution in multimedia systems. *IEEE Transanctions on Multimedia*, 10(7):1288–1298, November.

Atrey, Pradeep K., M. Anwar Hossain, Abdulmotaleb El Saddik, and Mohan S. Kankanhalli. 2010. Multimodal fusion for multimedia analysis: A survey. *Multimedia Systems*, 16(6):345–379.

Atrey, Pradeep K., Mohan S. Kankanhalli, and Ramesh Jain. 2006. Information as-

similation framework for event detection in multimedia surveillance systems. *Multimedia Systems*, 12(3):239–253.

Barbu, Costin, Raja Tanveer Iqbal, and Jing Peng. 2005. Classifier fusion using shared sampling distribution for boosting. In *International Conference on Data Mining*, pages 34–41.

Bellman, Richard E. 1961. *Adaptive control processes - A guided tour*. Princeton University Press.

Bensusan, Hilan and Christophe Giraud-Carrier. 2000. Casa batlo is in passeig de gracia or landmarking the expertise space. In *Proceedings of the ECML'2000 workshop on Meta-Learning: Building Automatic Advice Strategies for Model Selection and Method Combination*, pages 29–47, June.

Berger, Adam L., Stephen A. Della Pietra, and Vincent J. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22:39–71.

Beyer, Kevin, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. 1999. When is "nearest neighbor" meaningful? In *International Conference on Database Theory*, pages 217–235.

Bi, Yaxin, Jiwen Guan, and David Bell. 2008. The combination of multiple classifiers using an evidential reasoning approach. *Artificial Intelligence*, 172(15):1731–1751.

Blum, Avrim and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *Annual conference on Computational learning theory*, pages 92–100.

Bordley, Robert F. 1982. A multiplicative formula for aggregating probability assessments. *Management Science*, 28(10):1137–1148.

Braun, Jerome J. 2000. Dempster-shafer theory and bayesian reasoning in multisensor

data fusion. In *Sensor Fusion: Architectures, Algorithms, and Applications IV (SPIE)*, volume 4051, pages 255–266, April.

Brazdil, Pavel B. and Carlos Soares. 2000. Ranking classification algorithms based on relevant performance information. In *Proceedings of the ECML'2000 workshop on Meta-Learning: Building Automatic Advice Strategies for Model Selection and Method Combination*.

Breiman, Leo. 1996. Bias, variance, and arcing classifiers. Technical report, University of California Berkeley.

Cai, Na, Ming Li, Shouxun Lin, Yongdong Zhang, and Sheng Tang. 2007. Ap-based adaboost in high level feature extraction at trecvid. *International Conference on Pervasive Computing and Applications*, pages 194–198.

Campbell, John Y. and Luis M. Viceira. 2002. *Strategic Asset Allocation*. Oxford University Press.

Chandra, Siddharth and William G. Shadel. 2007. Crossing disciplinary boundaries: Applying financial portfolio theory to model the organization of the self-concept. *Journal of Research in Personality*, 41(2):346 – 373.

Chang, Chih-Chung and Chih-Jen Lin, 2001. *LIBSVM: a library for support vector machines*. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Chen, Jian-Guo and Nirwan Ansari. 1998. Adaptive fusion of correlated local decisions. *IEEE Transactions on Systems, Man, and Cybernetics*, 28(2):276–281.

Chen, Qi and Uwe Aickelin. 2006. Anomaly detection using the dempster-shafer method. In *International Conference on Data Mining*, pages 232–240.

Christel, Michael G., Chang Huang, and Neema Moraveji. 2004. Exploiting multiple modalities for interactive video retrieval. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1032–1035.

Conroy, Michael E. 1975. The concept and measurement of regional industrial diversification. *Southern Economic Journal*, 41(3):492–505.

Contreras, F. Martinez, C. Orrite Urunuela, and J. Martinez del Rincon. 2009. Adaboost multiple feature selection and combination for face recognition. In *Iberian Conference on Pattern Recognition and Image Analysis*, pages 338–345.

Cremer, Frank, Eric den Breejen, and Klamer Schutte. 1998. Sensor data fusion for anti-personnel land-mine detection. In *EuroFusion*, pages 55–60.

Dailey, Daniel J., Patricia Harn, and Po-Jung Lin. 1996. Its data fusion. Technical report, Washington State Transportation Center (TRAC), April.

Dasarathy, B. V. 1994. *Decision Fusion*. Computer Society Press.

Datta, Ritendra, Dhiraj Joshi, Jia Li, and James Z. Wang. 2006. Studying aesthetics in photographic images using a computational approach. In *European Conference on Computer Vision*, pages 7–13.

Datta, Ritendra, Jia Li, and James Z. Wang. 2008. Algorithmic inferencing of aesthetics and emotion in natural images: An exposition. In *IEEE International Conference on Image Processing*, pages 105–108.

Duin, Robert P.W. 2002. The combining classifier: to train or not to train? In *International Conference on Pattern Recognition*, pages II: 765–770.

Fassinut-Mombot, Bienvenu and Jean-Bernard Choquel. 2004. A new probabilistic and entropy fusion approach for management of information sources. *Information Fusion*, 5(1):35–47.

Foresti, Gian Luca and Lauro Snidaro. 2002. A distributed sensor network for video surveillance of outdoor environments. In *IEEE International Conference on Image Processing*, pages 525–528.

Freund, Yoav and Robert E. Schapire. 1996. Experiments with a new boosting algorithm. In *International Conference on Machine Learning*, pages 148–156.

Freund, Yoav and Robert E. Schapire. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55:23–37.

Gabrys, Bogdan and Dymitr Ruta. 2006. Genetic algorithms in classifier fusion. *Applied Soft Computing*, 6(4):337–347.

Gao, Sheng, Joo-Hwee Lim, and Qibin Sun. 2007. An integrated statistical model for multimedia evidence combination. In *ACM International Conference on Multimedia*, pages 872–881.

Geng, Xin, Kate Smith-Miles, Liang Wang, Ming Li, and Qiang Wu. 2010. Context-aware fusion: A case study on fusion of gait and face for human identification in video. *Pattern Recognition*, 43(10):3660–3673.

Guironnet, M., D. Pellerin, and M. Rombaut. 2005. Video classification based on low-level feature fusion model. In *European Signal Processing Conference*, pages 118–121.

Guo, Guodong and HongJiang Zhang. 2001. Boosting for fast face recognition. Technical Report MSR-TR-2001-16, Microsoft Research (MSR).

Guo, Wei, Qingwen Qi, Lili Jiang, and Ping Zhang. 2008. A new method of sar image target recognition based on adaboost algorithm. *IEEE International Geoscience and Remote Sensing Symposium*, 3:1194–1197.

Helmbold, David P., Robert E. Schapire, Yoram Singer, and Manfred K. Warmuth. 1998. On-line portfolio selection using multiplicative updates. *Mathematical Finance*, 8(4):325–347.

Heskes, Tom. 1998. Selecting weighting factors in logarithmic opinion pools. *Advances in Neural Information Processing Systems*, pages 266–272.

Hubbard, Douglas W. 2007. *How to Measure Anything: Finding the Value of "Intangibles" in Business*. Wiley.

Iyengar, G., H. J. Nock, and C. Neti. 2003a. Audio-visual synchrony for detection of monologues in video archives. In *International Conference on Multimedia and Expo*, pages 329–332.

Iyengar, G., HJ Nock, and C. Neti. 2003b. Discriminative model fusion for semantic concept detection and annotation in video. In *ACM International Conference on Multimedia*, pages 255–258.

Jagannathan, Ravi and Zhenyu Wang. 1996. The conditional capm and the cross-section of expected returns. *The Journal of Finance*, 51(1):3–53.

Jambor, Tamas and Jun Wang. 2010. Goal-driven collaborative filtering: A directional error based approach. In *European Conference on Information Retrieval*, pages 407–419.

Järvelin, Kalervo and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4):422–446, October.

Jasinschi, R. S., N. Dimitrova, T. McGee, L. Agnihotri, J. Zimmerman, D. Li, and J. Louie. 2002. A probabilistic layered framework for integrating multimedia content and context information. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 2057–2060.

Joshi, Dhiraj, Milind R. Naphade, and Apostol Natsev. 2007. Semantics reinforcement and fusion learning for multimedia streams. In *International Conference on Image and Video Retrieval*, pages 309–316.

Kay, Rakowsky Uwe. 2007. Fundamentals of the dempster-shafer theory and its applications to system safety and reliability modelling. *Reliability: Theory & Applications*, December.

Keller, James M., Paul D. Gader, and Charles W. Caldwell. 1995. Principle of least

commitment in the analysis of chromosome images. *Applications of Fuzzy Logic Technology II*, 2493(1):178–186.

Keller, Jörg, Iain Paterson, and Helmut Berrer. 2000. An integrated concept for multi-criteria-ranking of data-mining algorithms. In *Proceedings of the ECML'2000 workshop on Meta-Learning: Building Automatic Advice Strategies for Model Selection and Method Combination*.

Kittler, Josef, Mohamad Hatef, Robert P. W. Duin, and Jiri Matas. 1998. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20:226–239.

Kludas, J., E. Bruno, and S. Marchand-Maillet. 2007. Information fusion in multimedia information retrieval. *Adaptive Multimedial Retrieval: Retrieval, User, and Semantics*, pages 147–159.

Kohlas, Jürg and Paul-Andre Monney. 1995. *A Mathematical Theory of Hints: An Approach to the Dempster-Shafer Theory of Evidence*, volume 425 of *Lecture Notes in Economics and Mathematical Systems*. Springer.

Koks, Don and Subhash Challa. 2003. An introduction to bayesian and dempster-shafer data fusion. Technical Report DSTO-TR-1436, Defence Science and Technology Organisation.

Koval, O., S. Voloshynovskiy, and T. Pun. 2007. Error exponent analysis of person identification based on fusion of dependent/independent modalities. *SPIE-IS&T Electronic Imaging*.

Lee, Jong-Seok and Cheol Hoon Park. 2008. Adaptive decision fusion for audio-visual speech recognition. *Speech Recognition, Technologies and Applications*, pages 275–296.

Li, Francis C., Anoop Gupta, Elizabeth Sanocki, Li-wei He, and Yong Rui. 2000.

Browsing digital video. In *SIGCHI conference on Human factors in computing systems*, pages 169–176, New York, NY, USA. ACM.

Li, Ming, Yantao Zheng, Shouxun Lin, Yong-Dong Zhang, and Tat-Seng Chua. 2009. Multimedia evidence fusion for video concept detection via OWA operator. *International Conference on MultiMedia Modeling*, 5371:208–216.

Llinas, J., C. Bowman, G. Rogova, A. Steinberg, E. Waltz, and F. White. 2004. Revisiting the jdl data fusion model ii. In *International Conference on Information Fusion*, volume 2, pages 1218–1230.

Lucey, Simon, Sridha Sridharan, and Vinod Chandran. 2001. Improved speech recognition using adaptive audio-visual fusion via a stochastic secondary classifier. *International Symposium on Intelligent Multimedia, Video and Speech Processing*, pages 551–554, May.

Machajdik, Jana and Allan Hanbury. 2010. Affective image classification using features inspired by psychology and art theory. In *ACM International Conference on Multimedia*, pages 83–92.

Maghooli, K. and M. S. Moin. 2004. A new approach on multimodal biometrics based on combining neural networks using adaboost. In *Biometric Authentication*, pages 332–341.

Manyika, J. and H. Durrant-Whyte. 1994. *Data Fusion and Sensor Management: A Decentralized Information-Theoretic Approach*. Prentice Hall.

Markowitz, Harry. 1952. Portfolio selection. *The Journal of Finance*, 7(1):77–91.

Massaro, D.W. 2001. Auditory visual speech processing. *European Conference on Speech Communication and Technology*, pages 1153–1156.

Movellan, Javier R. and Paul Mineiro. 1998. Robust sensor fusion: Analysis and application to audio visual speech recognition. *Machine Learning*, 32:85–100, August.

Muneesawang, Paisarn, Ling Guan, and Tahir Amin. 2010. A new learning algorithm for the fusion of adaptive audio-visual features for the retrieval and classification of movie clips. *Signal Processing Systems*, 59(2):177–188.

Neti, Chalapathy, Benoît Maison, Andrew W. Senior, Giridharan Iyengar, P. Decuetos, Sankar Basu, and Ashish Verma. 2000. Joint processing of audio and visual information for multimedia indexing and human-computer interaction. In *International Conference on Computer-Assisted Information Retrieval*, pages 294–301.

Ngo, Chong-Wah, Yu-Gang Jiang, Xiao-Yong Wei, Feng Wang, Wanlei Zhao, Hung-Khoon Tan, and Xiao Wu. 2007. Experimenting VIREO-374: Bag-of-visual-words and visual-based ontology for semantic video indexing and search. *NIST TRECVID workshop*.

Nigam, K., J. Lafferty, and A. McCallum. 1999. Using maximum entropy for text classification. In *IJCAI workshop on machine learning for information filtering*, volume 1, pages 61–67.

Nilsson, Nils J. 1965. *Learning machines; foundations of trainable pattern-classifying systems*. McGraw-Hill New York.

Nocedal, Jorge and Stephen Wright. 2006. *Numerical Optimization*. Springer, 2nd edition.

Papandreou, George, Athanassios Katsamanis, Vassilis Pitsikalis, and Petros Maragos. 2009. Adaptive multimodal fusion by uncertainty compensation with application to audiovisual speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(3):423–435.

Parikh, Devi, Min T. Kim, Joseph Oagaro, Shreekanth Mandayam, and Robi Polikar. 2004. Combining classifiers for multisensor data fusion. In *International Conference on Systems, Man and Cybernetics*, pages 1232–1237.

Pfahringer, Bernhard, Hilan Bensusan, and Christophe Giraud-Carrier. 2000. Meta-

learning by landmarking various learning algorithms. In *International Conference on Machine Learning*, pages 743–750, June.

Pickering, M. J., S. M. Ruger, and D. Sinclair. 2002. Video retrieval by feature learning in key frames. In *International Conference on Image and Video Retrieval*, pages 309–317.

Poh, Norman and Samy Bengio. 2005. How do correlation and variance of base-experts affect fusion in biometric authentication tasks? *IEEE Transactions on Signal Processing*, 53(11):4384–4396.

Polikar, Robi. 2006. Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine*, 6(3):21–45.

Polikar, Robi, L. Upda, S. S. Upda, and Vasant Honavar. 2001. Learn++: an incremental learning algorithm for supervised neural networks. *IEEE Transactions on Systems, Man, and Cybernetics*, 31(4):497–508.

Punska, Olena. 1999. Bayesian approaches to multi-sensor data fusion. Master's thesis, University of Cambridge.

Reddy, Bakkama Srinath. 2007. Evidential reasoning for multimodal fusion in human computer interaction. Master's thesis, University of Waterloo.

Ross, Arun and Rohin Govindarajan. 2005. Feature level fusion using hand and face biometrics. *SPIE Conference on Biometric Technology for Human Identification*, 5779(II):196–204.

Ruta, Dymitr and Bogdan Gabrys. 2001. Application of the evolutionary algorithms for classifier selection in multiple classifier systems with majority voting. In *International Workshop on Multiple Classifier Systems*, pages 399–408.

Ruta, Dymitr and Bogdan Gabrys. 2005. Classifier selection for majority voting. *Information Fusion*, 6(1):63 – 81.

Sanderson, C. and Kuldip K. Paliwal. 2004. Identity verification using speech and face information. *Digital Signal Processing*, 14(5):449–480.

Schapire, Robert E. 1990. The strength of weak learnability. *Machine Learning*, 5(2):197–227.

Shafer, Glenn. 1976. *A Mathematical Theory of Evidence*. Princeton University Press.

Shafer, Glenn, 1992. *Encyclopedia of Artificial Intelligence*, chapter The Dempster-Shafer theory, pages 330–331. Wiley, 2 edition.

Singh, R., M. Vatsa, A. Noore, and S.K. Singh. 2006. Ds theory based fingerprint classifier fusion with update rule to minimize training time. *IEICE Electronics Express*, 3(20):429–435.

Smeaton, Alan F., Paul Over, and Wessel Kraaij. 2009. High-Level Feature Detection from Video in TRECVid: a 5-Year Retrospective of Achievements. In *Multimedia Content Analysis, Theory and Applications*. pages 151–174.

Smets, Philippe. 1998. The transferrable belief model for quantified belief representation. In *Handbook of Defeasible Reasoning and Uncertainty Management Systems, Volume 1: Quantified Representation Uncertainty and Imprecision*. Kluwer Academic Publishers, pages 267–301.

Snidaro, Lauro, Gian Luca Foresti, Gian Luca, Ruixin Niu, and Pramod K. Varshney. 2004. Sensor fusion for video surveillance. In *International Conference on Information Fusion*, pages 739–746.

Snoek, Cees, Marcel Worring, and Arnold W. M. Smeulders. 2005. Early versus late fusion in semantic video analysis. In *ACM International Conference on Multimedia*, pages 399–402.

Srinivas, Nisha, Kalyan Veeramachaneni, and Lisa Ann Osadciw. 2009. Fusing correlated data from multiple classifiers for improved biometric verification. In *International Conference on Information Fusion*.

Stone, M. 1961. The opinion pool. *The Annals of Mathematical Statistics*, 32(4):1339–1342.

Tan, Li, Yuanda Cao, Minghua Yang, and Jiong Yu. 2009. A novel fusion method for semantic concept classification in video. *Journal of Software*, 4(9):968–975.

Tax, David M.J., Martijn Van Breukelen, Robert P.W. Duin, and Josef Kittler. 2000. Combining multiple classifiers by averaging or by multiplying? *Pattern Recognition*, 33:1475–1485.

Telmoudi, A. and S. Chakhar. 2004. Data fusion application from evidential databases as a support for decision making. *Information and Software Technology*, 46(8):547–555.

Tieu, K. and P. Viola. 2004. Boosting image retrieval. *International Journal of Computer Vision*, 56(1-2):17–36, January.

Tipping, Michael E. 1999. The relevance vector machine. *Advances in Neural Information Processing Systems*, pages 652–658.

Valiant, L. G. 1984. A theory of the learnable. In *ACM symposium on Theory of computing*, pages 436–445.

Černý, Aleš. 2003. *Mathematical Techniques in Finance: Tools for Incomplete Markets*. Princeton University Press.

Vilalta, Ricardo and Youssef Drissi. 2002. A perspective view and survey of meta-learning. *Artificial Intelligence Review*, 18:77–95.

Wang, Jun and Jianhan Zhu. 2009. Portfolio theory of information retrieval. *ACM Special Interest Group on Information Retrieval Conference*, pages 115–122.

Wang, Meng, Xian-Sheng Hua, Xun Yuan, Yan Song, and Li-Rong Dai. 2007. Optimizing multi-graph learning: towards a unified video annotation scheme. In *ACM International Conference on Multimedia*, pages 862–871.

Wang, Meng, Linjun Yang, and Xian-Sheng Hua. 2009. Msra-mm: Bridging research

and industrial societies for multimedia information retrieval. TechReport MSR-TR-2009-30, Microsoft Research (MSR).

Wang, Xiangyu and Mohan S. Kankanhalli. 2010a. Multifusion: A boosting approach for multimedia fusion. *ACM Transactions on Multimedia Computing, Communications and Applications*, 6(4), November.

Wang, Xiangyu and Mohan S. Kankanhalli. 2010b. Portfolio theory of multimedia fusion. In *ACM International Conference on Multimedia*, pages 723–726.

Wu, Shengli and Fabio Crestani. 2002. Data fusion with estimated weights. In *International conference on Information and knowledge management*, pages 648–651.

Wu, Shengli and McClean. 2005. Data fusion with correlation weights. In *European Conference on Information Retrieval*, pages 275–286.

Wu, Yi, Edward Y. Chang, Kevin Chen-Chuan Chang, and John R. Smith. 2004. Optimal multimodal fusion for multimedia data analysis. In *ACM International Conference on Multimedia*, pages 572–579.

Xue, Feng and Xiaoqing Ding. 2006. 3D+2D face localization using boosting in multimodal feature space. In *International Conference on Pattern Recognition*, pages 499–502.

Yager, R.R. 1988. On ordered weighted averaging aggregation operators in multicriteria decisionmaking. *IEEE Transactions on Systems, Man, and Cybernetics*, 18(1):183–190.

Yan, Rong and Alexander G. Hauptmann. 2003. The combination limit in multimedia retrieval. In *ACM International Conference on Multimedia*, pages 339–342.

Yanagawa, Akira, Shih-Fu Chang, Lyndon Kennedy, and Winston Hsu. 2007. Columbia university's baseline detectors for 374 lscom semantic visual concepts. TechReport 222-2006-8, Columbia University.

Yanagawa, Akira, Winston Hsu, and Shih-Fu Chang. 2006. Brief descriptions of vi-

sual features for baseline trecvid concept detectors. Technical report, Columbia University, July.

Yang, Linjun, Jiemin Liu, Xiaokang Yang, and Xian-Sheng Hua. 2007. Multi-modality web video categorization. In *ACM SIGMM International Workshop on Multimedia Information Retrieval*, pages 265–274.

Zhang, L., M. J. Li, and H. J. Zhang. 2002. Boosting image orientation detection with indoor vs. outdoor classification. In *IEEE Workshop on Applications of Computer Vision*, pages 95–99.

Zhang, Tong and Bin Yu. 2005. Boosting with early stopping: Convergence and consistency. *The Annals of Statistics*, 33:1538–1579.

Zhang, X., T. Sim, and X. Miao. 2008. Enhancing photographs with near infra-red images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8.

Zheng, Y. T., S. Y. Neo, T. S. Chua, and Q. Tian. 2008. Probabilistic optimized ranking for multimedia semantic concept detection via RVM. *International Conference on Image and Video Retrieval*, pages 161–168.

Zhu, Qiang, Mei-Chen Yeh, and Kwang-Ting Cheng. 2006. Multimodal fusion using learned text concepts for image categorization. In *ACM International Conference on Multimedia*, pages 211–220.