# A RELIEF-Based Modality Weighting Approach for Multimodal Information Retrieval

Turgay Yilmaz*
Dept.of Computer Engineering
Middle East Technical Univ.
06531, Ankara, Turkey
turgay@ceng.metu.edu.tr

Elvan Gulen
Dept.of Computer Engineering
Middle East Technical Univ.
06531, Ankara, Turkey
elvan@ceng.metu.edu.tr

Adnan Yazici
Dept.of Computer Engineering
Middle East Technical Univ.
06531, Ankara, Turkey
yazici@ceng.metu.edu.tr

Masaru Kitsuregawa
Institute of Industrial Science
University of Tokyo
Tokyo 153-8505, Japan
kitsure@tkl.iis.u-tokyo.ac.jp

## ABSTRACT

Despite the extensive number of studies for multimodal information fusion, the issue of determining the optimal modalities has not been adequately addressed yet. In this study, a RELIEF-based multimodal feature selection approach (RELIEF-RDR) is proposed. The original RELIEF algorithm is extended for weaknesses in three major issues; multi-labeled data, noise and class-specific feature selection. To overcome these weaknesses, discrimination based weighting mechanism of RELIEF is supported with two additional concepts; representation and reliability capabilities of features, without an increase in computational complexity. These capabilities of features are exploited by using the statistics on dissimilarities of training instances. The experiments conducted on TRECVID 2007 dataset validated the superiority of RELIEF-RDR over RELIEF.

## Categories and Subject Descriptors

H.3.1 [**Information Systems**]: Information Storage and Retrieval—*Content Analysis and Indexing*; I.5.2 [**Computing Methodologies**]: Pattern Recognition—*Design Methodology*

## General Terms

Algorithms, Experimentation, Performance

## Keywords

RELIEF, Feature Weighting, Multimodal Information Fusion

*Turgay Yilmaz is currently with the Institute of Industrial Science, University of Tokyo, Japan.


*ICMR* '12, June 5-8, Hong Kong, China


## 1. INTRODUCTION

Retrieval of multimedia data, bases on the semantic content in multimedia data. To understand the semantic content effectively, the nature of the multimedia data should be noticed and the information contained should be totally used. The multimedia data has a complex structure containing multimodal information (i.e. audial, visual and textual modalities). Regarding that the noise in sensed data, non-universality and performance upper bound of any single modality prevent relying on a single modality [24]; it is obvious that employing multiple modalities effectively will increase the retrieval performance. In order to use the multiple modalities effectively, a selection (or weighting) on the available modalities should be applied, which is one of the important issues that have not been adequately addressed yet in the information fusion domain [20, 24, 35].

The work done so far on using multiple modalities are in three groups: (i) using all features/modalities by averaging them (ii) performing an empirical selection and (iii) determining effectiveness of each multimodal feature with a selection algorithm. The first two one are simplistic approaches; the first one behaves all features equally-likely although any of the features can be non-informative or redundant, whereas the second approach requires an empirical observation and manual selection on the observation. Besides, the third direction requires design of an efficient feature selection / weighting algorithm, which proposes a polynomial time heuristic for the NP-hard selection problem. Aligned to the third direction, we would like to investigate the effect of class-specific feature selection [34, 36] on modality aggregation in multimedia retrieval. The idea of class-specific features requires selecting a possible different feature subset for each class, instead of a class-common case, where a single feature subset is selected for all the classes, as almost all of the feature selection approaches apply.

Besides, one important issue in modality selection is the dependency among the features. The success of the combination result can be better than all of the inputs if only the inputs are complementary (independent) [15, 16]. Thus, as the multimodal feature selection algorithm, we propose a RELIEF-based weighting approach, with a motivation that RELIEF [13] is one of the most successful algorithms for fea-

ture selection, and known to be a simple and effective way of correctly estimating the quality of features, even in problems with strong dependencies between features. In addition, to the best of our knowledge, there exists no usage of RELIEF algorithm for multimodal feature selection. We start with RELIEF-F, which is the multi-class extension of RELIEF, and extend the algorithm due to three crucial aspects:

(i) RELIEF cannot perform well when the training samples are multi-labeled. RELIEF estimates weights of features according to their ability to discriminate between different classes by analyzing the distances of the samples with their neighboring same-class and different-class instances. However, having multi-labeled samples causes the algorithm not to discriminate between classes effectively.

(ii) RELIEF is not noise-tolerant. Similar to the multi-label issue, having noise in the samples prevents a correct discrimination between classes.

(iii) Regular use of RELIEF does not provide a class-specific solution. Although it may be possible to generate class-specific weights if the algorithm is executed separately for each class, doing so makes the process more complicated and requires extra effort as well as a motivation for class-specific selection.

Considering that the multimedia data usually has a multi-labeled structure and can contain a high amount of noise, in order to overcome above given problems of RELIEF-F for multimedia data, we suggest a new extended RELIEF algorithm using the representation and reliability characteristics of the features as well as their discrimination capabilities, namely RELIEF algorithm based on Representation, Discrimination and Reliability (RELIEF-RDR). Another important specialty of RELIEF-RDR is that it performs a class-specific feature selection directly. One last difference with RELIEF is that the given three characteristics of features are calculated based on the statistics of distances between the training instances, instead of the distances directly. The average distance of the samples to themselves for each class and corresponding standard deviations are employed as the representative characteristics, and the correctness ratios of features for each class are used as the reliability characteristics. For the discriminative property, we calculate the distance between the means of classes. Having such a preference decreases the effect of noise.

The computational complexity of the proposed algorithm is not worse than the original algorithm. The multimodal retrieval tests performed on the TRECVID 2007 dataset has shown that the proposed RELIEF-RDR algorithm generates better feature weights than RELIEF-F algorithm. Also it has been observed that RELIEF-RDR obtains better performance than a class-common exhaustive search.

The remainder of this paper is organized as follows: In Section 2, an overview on modality selection in information fusion, feature selection methods and RELIEF algorithms is given. In Section 3, the proposed algorithm is given in detail. In Section 4, the empirical results and the evaluations are presented. Lastly, in Section 5, some conclusions are drawn.

# 2. RELATED WORK

## 2.1 Multimodal Feature Selection in Information Fusion

In the information fusion literature, a big majority of studies prefer using all available modalities or employing an empirical weighting scheme [3,9,16,32]. Such methods assumes independency of the inputs and benefits from the simplicity in calculation and the robustness in estimating the evidence [11]. However, it is not always useful to combine all inputs considering that the dependent inputs hurt the information fusion performance [16]. An important issue in fusion is that the success of the fusion result can be better than all of the inputs if only the inputs are complementary (independent) [15, 16]. Yet, there are several studies proposing methods to use different combinations of available modalities. Some of the recent approaches in the fusion literature can be listed as follows: performing feature selection or transformation by finding principal/independent components [16, 35], selecting the most coherent and less complex features according to the heterogeneity of features [15], calculating the information gain obtained [1, 12], defining quality and reliability metrics on features [24, 31]. However, as indicated by Atrey et al. [20], these studies are still very few and there are a lot more can be done in this aspect. Furthermore, these methods have common weakness: The selection process is usually class-common, which means, the same set of features are preferred for all classes. Considering the idea that different features can be more effective for different classes [23, 34], using class-specific feature weights can be more beneficial.

## 2.2 Feature Selection Approaches

In addition to the above given methodologies, the *Feature Selection* studies in *Pattern Recognition* literature provide much more approaches for selection, despite the underuse of them in modality selection for fusion. Existing feature selection methods in the literature are categorized as filter or wrapper methods. Filter methods assess the relevance of features by looking only at the intrinsic properties of the data, whereas in wrapper methods the performance of a learning algorithm is used to evaluate the fitness of the feature subsets in the feature space. Filter methods are usually computationally better than wrapper methods, however wrapper methods provide more optimal solutions. Some well-known filter methods are Information Gain [8], Gain Ratio [25], Correlation based feature selection (CFS) [7] and RELIEF [13]. Also, some well-known wrapper methods are as follows: Sequential Forward selection (SFS) [14], Sequential Backward elimination (SBE) [14], Plus q take-away r [5], Simulated Annealing and Genetic Algorithms. A more detailed discussion of these methods can be found in [6, 10, 28].

## 2.3 RELIEF Algorithms

Among the available feature selection and weighting methods, the RELIEF algorithm [13] is one of the most successful ones. It is a simple and effective way of selection [4], and does not make a conditional independence assumption for features, as many other feature selection methods do. RELIEF can correctly estimate the quality of features with dependencies [27]. The key idea of RELIEF is to estimate weights for each feature according to their ability to discriminate between neighboring training samples by iterating through randomly selected instances in the training space. The algorithm for basic RELIEF is given in Algorithm 1. The weight formula in Line 9 exploits the discrimination capability. The algorithm selects a random sample $R$, one Near-Hit $H$ (nearest neighbor with the same class with the random sample) and one Near-Miss $M$ (nearest neighbor

**Algorithm 1:** Basic RELIEF Algorithm

**Input**: features $F$, number of iterations $m$, training instances with feature values and classes
**Output**: the weight vector W of estimations for the qualities of features

1 **begin**
2  **foreach** *feature* $f \in F$ **do**
3  set weight $W[f] := 0$;
4  **end foreach**
5  **for** $i := 1$ **to** $m$ **do**
6  randomly select an instance $R$;
7  find nearest hit $H$ and nearest miss $M$;
8  **foreach** *feature* $f \in F$ **do**
9  $W[f] := W[f] - \frac{diff(f,R,H)}{m} + \frac{diff(f,R,M)}{m}$;
10  **end foreach**
11  **end for**
12 **end**

with a different class with the random sample) and distances between them are calculated. Distance between instances in different classes provides discrimination, so $diff(f, R, M)$ increases the weight. Inversely, distance between instances with the same class prevents discrimination, so $diff(f, R, H)$ decreases the weight.

Considering several deficiencies of basic RELIEF algorithm, Kononenko [17] proposes several extensions for RELIEF: RELIEF-A uses $k$ nearest neighbors instead of one and averages the contribution of k nearest instances in order to prevent the effect of noisy instances; RELIEF-B, RELIEF-C and RELIEF-D extends the use of $diff(feature, Instance_1, Instance_2)$ in order to handle incomplete dataset; RELIEF-E and RELIEF-F improves the weight update function for multi-class problems. Other well-known extensions for RELIEF are as follows: Sikonja et al. [26] proposes RRELIEF-F for handling regression problems. In [30], Sikonja proposes using k-d trees for the selection of nearest neighbors in order to decrease the computation complexity of the RELIEF algorithm. In [33], Sun introduces Iterative RELIEF (I-RELIEF), which uses Expectation-Maximization algorithm in order to eliminate outlier data. Also, Liu et al. [18] try to eliminate outlier data and propose using selective sampling by means of a modified kd-tree instead of random sampling (at Line 6 in Algorithm 1).

Among the available extensions of RELIEF algorithm, RELIEF-F is the mostly utilized one. In RELIEF-F algorithm, k nearest misses for each class and k nearest hits are used, in order to handle multi-class problem. Selection of $k$ hits and misses provides greater robustness of the algorithm concerning noise in the dataset. However, such action cannot prevent the effect of noise, when larger values of $k$ is used. It has been observed that, in a setting with dependent features, larger values of $k$ prevents RELIEF-F to distinguish the informative features [17, 27]. The power of RELIEF-F is its ability to exploit information locally, taking the context into account by means of distances between the instances. However, for larger number of nearest neighbors, the perspective changes from local to global. Then, for any feature in the feature space, regardless of its informativeness, the positive and negative updates are equiprobable [27] and such updates lead to weights near zero.

## 2.4 Complexity Analysis

The feature selection / weighting problem is known as NP-hard, in terms of the number of features $f$. An exhustive search for generating all possible subset requires $O(p^f)$ actions, where $p$ is the number of assignable weights ($p = 2$ for binary selection). If we consider an evaluation for each of these subsets, the total complexity of the exhaustive search becomes $O(m{\cdot}n{\cdot}f{\cdot}p^f)$. Moreover, if a class-specific approach is applied, the total complexity becomes $O(m \cdot n \cdot f \cdot p^{c \cdot f})$.

Besides, RELIEF algorithms offer solutions in polynomial time. Complexity of the basic RELIEF algorithm is $O(m \cdot n \cdot f)$, considering that the most complex operation is the selection of the nearest hit and and miss instances since the distances between $R$ and the other training instances should be calculated for each feature, which requires $O(n \cdot f)$ comparisons. The complexity of RELIEF-F is also similar. Yet, a computationally better solution can be obtained by utilizing k-d trees for improving the nearest hit and miss selection process ($O(f \cdot n \cdot logn)$).

# 3. RELIEF-RDR

In order to benefit from the simplicity and effectiveness of RELIEF algorithms, we propose a RELIEF-based multimodal feature selection algorithm. Moreover the capability of RELIEF correctly estimating the quality of features with dependencies is another important concern, considering that the dependency among features is an important issue to be handled during feature selection. However we locate three crucial issues to be extended in RELIEF.

First issue is having multi-labeled data. Multimedia data is a multi-labeled one, which can contain more than one concepts for each keyframe or shot in any of the modalities contained, i.e. having both an *airplane* and a *mountain* in a visual scene. Also, different modalities can contain different concepts at the same instance, i.e. having an *explosion* sound in the audial modality and *military* related vehicles in the visual modality at the same moment of the video. As mentioned before, RELIEF estimates weights of features by using the discrimination capability of features between different classes. However, having multi-labeled samples prevents such discrimination.

Second issue is the effect of noise. In addition to the fact that the multimedia data have an expected internal noise, the way we model the multimedia data can create an artificial noise. Since the multimedia data is usually large –even huge–, some sub-sampling (i.e. using shots and keyframes instead of each particular frame) is done before processing it. Then, the extracted features represents only subsamples from the video, whereas the ground truth labels are based on the full content of the video. Such a situation makes the evaluation of features complicated and eventually some of the ground truth instances seem as noisy instances. Similar to the multi-label issue, having noise in the samples prevents a correct discrimination between classes. In addition, depending directly on the distances between training instances affects the performance of the algorithm negatively, considering the noisy instances.

To overcome these two issues, we introduce two new factors for the calculation of the weights, in addition to the discrimination capability used in RELIEF: the representation and reliability characteristics. So, our algorithm is named as RELIEF algorithm based on Representation, Discrimination and Reliability (RELIEF-RDR). Having additional components in the weight calculation makes RELIEF-RDR less dependent to discrimination capability, and provides better

estimations. Selection of the characteristics is also important; a good feature should represent the class as much as possible, and the results obtained with such feature should be reliable, as well as having a good discrimination between classes. In the proposed algorithm, the representative characteristics of features are obtained by using the average distance of the samples to themselves for each class and corresponding standard deviations. For the reliability of features, we use the correctness ratios obtained via the retrieval accuracies of features for each class. For the discrimination capability, we change the calculation and use the distance between the means of classes. As seen, for the calculation of weights, we prefer the statistics of distances between the training instances, instead of the distances directly. Such a preference helps to normalize the used distance values in the calculations and decrease the effect of noise, which is caused by noisy training samples.

The third issue we located is the class-specific feature selection idea. Feature selection methods usually propose solutions such that the resulting feature set is selected independent of the classes. However, using the same features for different types of concepts can yield unsatisfactory results. Specifically considering the multimodality in the multimedia data, it can be more convenient to detect different classes with different feature sets. For instance, for an *explosion* event, the audial modality is more useful whereas it is better to utilize visual modality for detecting a *mountain* occurrence. Similarly, it can be easier to recognize a *meeting* event by using both the visual and the audial modalities. As mentioned before, regular use of RELIEF does not provide a class-specific solution. By claiming that having class-specific feature weights obtains better accuracy results, we extend the algorithm in such a way that it generates class-specific weights.

## 3.1 The Algorithm

The algorithm for RELIEF-RDR is presented in Algorithm 2. Different from RELIEF-F, RELIEF-RDR does not calculate the weights by iteratively updating them. In RELIEF-RDR, firstly the distances between randomly selected instances and their hits/misses are collected, and held in distance matrices. Then the distance matrices are aggregated according to the instance classes, which gives mean and standard deviations of distances between instance classes. By using the mean and the standard deviation matrices, weights of features for each class are calculated separately.

The algorithm begins with initializing four arrays: $W$, $D$, $Mean$, $StdDev$. $W$ is the feature vectors for each class. $D$ array is for distance matrices of each feature and holds the distances between randomly selected instances and their selected hits/misses. $Mean$ and $StdDev$ arrays are used for holding the averages and standard deviation of distances between each class pairs, after the aggregation on $D$.

After initializations, the iterative instance selection loop appears. Through an iteration of $m$ times, a random instance $R$ is selected among the training set. Then $k$ miss instances for each class and $k$ hit instances are selected according to the total distances with the $R$ instance. The total distance is calculated by using an Euclidian distance for all available features. After the selection, distances between $R$ and each hit instance ($diff(f, R, H_j)$) according to each single feature are inserted into the $D$. Also, similarly the distances ($diff(f, R, M_j(C))$) are inserted into $D$.

**Algorithm 2:** RELIEF-RDR Algorithm

**Input**: features $F$, number of iterations $m$, number of nearest selections $k$, number of classes $c$, training instances with feature values and classes
**Output**: the weight vector W of estimations for the qualities of features

**begin**
  init. weight vectors $W$: array of $[c][size(F)]$;
  init. dist.matrices $D$: array of $[size(F)][m][k \cdot c]$;
  init. mean matrices $Mean$: array of $[size(F)][c][c]$;
  init. std.dev.matrices $StdDev$: array of $[size(F)][c][c]$;
  **for** $i := 1$ **to** $m$ **do**
    randomly select an instance $R$;
    find nearest $k$ hits $H_j$ and nearest misses $M_j(C)$ of each class;
    **foreach** *nearest hit* $H_j$ **do**
      **foreach** *feature* $f \in F$ **do**
        $D[f][i][j] = diff(f, R, H_j)$;
      **end foreach**
    **end foreach**
    **foreach** *class* $C \neq class(R)$ **do**
      **foreach** *nearest miss* $M_j(C)$ **do**
        **foreach** *feature* $f \in F$ **do**
          $D[f][i][j] = diff(f, R, M_j(C))$;
        **end foreach**
      **end foreach**
    **end foreach**
  **end for**
  performAggregation($D$, $Mean$, $Std$);
  **foreach** *class* $C$ **do**
    **foreach** *feature* $f \in F$ **do**
      $MeanOfClass = Mean[f][C][C]$;
      $StdOfClass = Std[f][C][C]$;
      $MeanDist$=findMeanDist($Mean[f][C][C]$, $Mean[f][C]$);
      $CorrRatio$=findCorrRatio($Mean[f][C][C]$, $Mean[f][C]$);
      $W[C][f] = \frac{1-MeanOfClass}{StdOfClass} \cdot MeanDist^v \cdot CorrRatio$;
    **end foreach**
  **end foreach**
**end**

After finishing the iterative loop and filling the distance matrices, the matrices are aggregated both column-wise and row-wise according to the classes of the instances. Thus, average distances between class pairs are obtained as well the standard deviations, and hold in the mean and standard deviation matrices for each feature, the $Mean$ and the $StdDev$. After calculation of the $Mean$ and the $StdDev$, the weight calculation of each feature is performed for each class. As introduced before, weight formulation of RELIEF-RDR includes four parameters.

**Mean of Class:** $MeanOfClass$ is the average distance value of a class to itself, for a particular feature $f$. For a particular class, the features with lower distance values represent the class better. Thus, the weight of a feature is inversely proportional to the mean of the class.

**Standard Deviation of Class:** For any class, a feature with small standard deviation entails close instance-to-instance distance values within the class. Such a feature can be considered as better. Thus, the weight of a feature is inversely proportional to $StdOfClass$ of an image class.

**Standard Mean Distance to Other Classes:** To calculate the $MeanDist$, the distances of a class to other classes are used. The idea is similar to the RELIEF weight update formulation.

$$\begin{aligned} MeanDist(C,f) &= \sqrt{\frac{\sum_{C' \neq C} (A-B)^2}{c}} \qquad (1) \\ A &= Mean[f][C][C] \\ B &= Mean[f][C][C^{'}] \end{aligned}$$

where $c$ is the number of classes. This calculation gives us the average distance of an image class $C$ to all other classes. Thus, having a greater distance means better discrimination among all classes, which means that the weight is directly proportional to $MeanDist$.

**Correctness Ratio:** It is important for a feature to give the lowest distance values for the instances in a class which is the same with the class of the query instances. Correctness ratio ($CorrRatio$) of a particular feature $f$ can be defined as what percentage of the means in a $Mean[f][C]$ vector are larger than the mean value of the class $C$ ($MeanOfClass$). As the correctness ratio decreases, the reliability of feature decreases, which means that the weight is directly proportional with the correctness ratio.

Considering the effects of the above parameters, the weight formulation in RELIEF-RDR of a particular feature $f$ for a particular class $C$ is calculated using the formula below[1]:

$$W[C][f] = \frac{1 - MeanOfClass}{StdOfClass} \cdot MeanDist^{v} \cdot CorrRatio \qquad (2)$$

## 3.2 Complexity Analysis

We assume that $f$ denotes number of features, $m$ denotes number of iterations, $k$ denotes number of nearest selections, $c$ denotes number of classes $c$ and $n$ denotes number of training instances. Considering the Algorithm 2, RELIEF-RDR includes two main loops and and an aggregation operation on the $D$ array.

First loop is used for iterating over m instances (Lines 6-21) and contains three operations; (1) selection of hits and misses in Line 8 ($O(n \cdot f)$), (2) inserting distances between $R$ and $k$ hits into the $D$ array in Lines 9-13 ($O(k \cdot f)$), inserting distances between $R$ and $k$ misses for each class into the $D$ array in Lines 14-20 ($O(c \cdot k \cdot f)$). Considering that $n > k$ and $n > c \cdot k$ are always true since k can have a maximum value of $(n-1)/c$; the complexity of one iteration in the first loop is equal to $O(n \cdot f)$ asymptotically. Then total complexity of the first loop is $O(m \cdot n \cdot f)$.

The aggregation operation in Line 22 requires a full traversal in $D$ array, so the cost is $O(m \cdot c \cdot k \cdot f)$.

Second loop (Lines 23-31) is used for weight calculation and contains two complex operations, which requires looping over all classes, in two nested loops of sizes $c$ and $f$: (1) calculation of standard mean distance in Line 27 ($O(c)$) and (2) calculation of correctness ratio in Line 28 ($O(c)$). Then, total complexity of the second loop becomes $O(c^2 \cdot f)$.

Thus, according to above given calculations, complexity of RELIEF-RDR is $O(m \cdot n \cdot f + m \cdot c \cdot k \cdot f + c^2 \cdot f)$. As mentioned above, $n > c \cdot k$ is always true. Also, $m > c$ should be true, since the algorithm implicitly makes an assumption that at least one instance should be selected from each class in order to calculate feature weights of each class. Then, the complexity of the algorithm becomes equal to $O(m \cdot n \cdot f)$ asymptotically, which is the same as the original RELIEF-F algorithm.

[1]the power $v$ of $MeanDist$ can be assumed as an experimental constant.

# 4. EMPIRICAL STUDY

In this section, we evaluate the proposed modality weighting approach for the semantic retrieval of multimedia data. For the retrieval task, the multimedia data is queried based on the semantic classes. First, retrieval for each single feature is performed, then the features are combined with a linear (weighted sum) combiner based late-fusion approach by using the weights generated and a multimodal retrieval is done.

## 4.1 Experimental Setup

### 4.1.1 Dataset

Evaluations on multimodal setting are based on the international video information retrieval benchmark TRECVID. In our tests, TRECVID 2007 is considered [22]. TRECVID 2007 corpus is composed of 100 hours of multilingual video, roughly equally divided into training and test sets. The training data comprises 110 videos and 30.6 GB, whereas the test data is 109 files and 29.2 GB. The annotations on the TRECVID 2007 dataset is provided in a multi-label manner, which means each shot can contain more than one label.

For shot segmentation, the outputs of common shot reference are used. The dataset contains 21,532 reference shots for training and 18,142 reference shots for test. In the experiments, we used the 20 semantic concepts which were selected in TRECVID 2007 evaluation. During the tests, the shots are considered as individual and independent documents, which means no contextual information or interaction is taken into account between shots.

Further details and a performance comparison of TRECVID 2007 participants can be found in [22].

### 4.1.2 Modalities

Considering a multimodal setting; visual, audial and textual features are extracted from the videos. For visual features, one key frame per shot is adopted and the middle frame for each shot is selected as the key frame. For audial features, entire audio of each shot is processed. For the textual features, the Automatic Speech Recognition and Machine Translation texts, which are provided by TRECVID, are employed. In the tests, we employ the following multimodal features: Color Layout, Region Shape and Edge Histogram features of MPEG-7 [19], Perceptual audio features (Zero Crossing Rate and Energy), Cepstral audio features (Mel-frequencies Cepstrum Coefficients, MFCC) and Term Frequency–Inverse Document Frequency (TF-IDF) weights.

The feature extraction and distance calculation tasks of visual features are performed by using the MPEG-7 reference software (eXperimentation Model, XM) [21]. For the feature extraction of audio features Yaafe toolbox [2] is utilized and distances are calculated by Euclidean distance. For the textual modality, the term frequency–inverse document frequency (TF-IDF) weights [29] are calculated as features. During calculation, no stop-word filtering or preprocessing is done. For the distance calculation, Cosine similarity metric is used.

**Table 1: Accuracy comparisons. The best result for each category is highlighted in bold.**

| | Color | Shape | Texture | Perceptual Audio | Cepstral Audio | Textual | Average | Exhaustive Search | RELIEF-F $k$=20 | RELIEF-RDR ($v$=3, $k$=1200) |
|---|---|---|---|---|---|---|---|---|---|---|
| Airplane | 0.112 | 0.037 | 0.093 | 0.068 | 0.040 | 0.023 | 0.063 | 0.185 | **0.377** | 0.135 |
| Animal | 0.066 | 0.078 | **0.106** | 0.071 | 0.078 | 0.059 | 0.086 | 0.069 | 0.068 | 0.066 |
| Boat_Ship | **0.112** | 0.071 | 0.098 | 0.049 | 0.058 | 0.043 | 0.086 | 0.087 | 0.109 | 0.104 |
| Car | 0.189 | 0.181 | 0.196 | 0.138 | 0.182 | 0.115 | 0.224 | 0.210 | 0.182 | **0.230** |
| Charts | 0.008 | 0.025 | 0.024 | 0.027 | 0.019 | **0.029** | 0.013 | 0.009 | 0.024 | 0.023 |
| Computer_TV-screen | 0.059 | 0.073 | **0.096** | 0.049 | 0.072 | 0.080 | 0.079 | 0.086 | 0.096 | 0.089 |
| Desert | **0.013** | 0.010 | 0.013 | 0.005 | 0.007 | 0.007 | 0.012 | 0.013 | 0.005 | 0.006 |
| Explosion_Fire | 0.011 | 0.026 | 0.030 | 0.013 | 0.015 | 0.010 | 0.025 | 0.014 | **0.031** | 0.028 |
| Flag-US | 0.005 | 0.002 | 0.002 | 0.001 | 0.003 | **0.013** | 0.002 | 0.002 | 0.001 | 0.001 |
| Maps | 0.020 | 0.021 | 0.034 | 0.030 | **0.051** | 0.032 | 0.033 | 0.016 | 0.030 | 0.029 |
| Meeting | 0.376 | 0.264 | 0.395 | 0.352 | 0.300 | 0.328 | 0.271 | **0.464** | 0.357 | 0.422 |
| Military | 0.034 | 0.024 | 0.031 | 0.023 | 0.010 | 0.015 | **0.042** | 0.038 | 0.025 | 0.041 |
| Mountain | 0.044 | 0.030 | 0.057 | 0.025 | 0.030 | 0.023 | 0.038 | 0.046 | **0.057** | 0.057 |
| Office | 0.130 | 0.083 | 0.135 | 0.083 | 0.117 | 0.140 | 0.090 | 0.184 | 0.023 | **0.213** |
| People-Marching | 0.054 | 0.046 | 0.067 | 0.027 | 0.048 | 0.020 | 0.075 | **0.094** | 0.077 | 0.089 |
| Police_Security | 0.050 | 0.030 | 0.032 | 0.039 | 0.026 | 0.036 | 0.033 | **0.065** | 0.049 | 0.049 |
| Sports | **0.048** | 0.039 | 0.036 | 0.026 | 0.037 | 0.036 | 0.041 | 0.035 | 0.015 | 0.040 |
| Truck | 0.078 | 0.101 | 0.118 | 0.070 | 0.085 | 0.052 | 0.104 | 0.092 | **0.118** | 0.101 |
| Waterscape_Waterfront | 0.161 | 0.076 | 0.139 | 0.066 | 0.083 | 0.071 | 0.142 | 0.155 | 0.161 | **0.173** |
| Weather | 0.036 | 0.002 | 0.003 | 0.003 | 0.003 | 0.001 | 0.005 | 0.004 | 0.002 | **0.070** |
| MAP | 0.080 | 0.061 | 0.085 | 0.058 | 0.063 | 0.057 | 0.073 | 0.093 | 0.090 | **0.098** |
| MAP Rank | 5 | 8 | 4 | 9 | 7 | 10 | 6 | 2 | 3 | **1** |
| Number of Best Scores | 3 | 0 | 2 | 0 | 1 | 2 | 1 | 3 | **4** | **4** |
| Mean Rank | 4.9 | 6.8 | 4.0 | 7.7 | 6.5 | 7.4 | 4.8 | 4.4 | 4.9 | **3.7** |

#### 4.1.3 Metrics

To measure the retrieval accuracy, *Average Precision(AP)* and *Mean Average Precision(MAP)* are used. The $AP$ is the sum of the precision at each relevant hit in the retrieved list, divided by the minimum between the number of relevant documents in the collection and the length of the list. Regarding the evaluation rules of TRECVID, AP is measured at 2000. $MAP$ is the $AP$ averaged over several image classes. In other words, the $AP$ of each image class is calculated separately, then the $MAP$ is found by averaging them.

#### 4.1.4 Comparison

For comparison, we evaluate our approach against (i) each single modality, (ii) simple averaging of all modalities (iii) exhaustive search and (iv) RELIEF-F. During comparison, not only the RELIEF-RDR weights, but also the four parameters of the CSF RELIEF-RDR are tested separately in order to see which one is more influential. By performing some initial tests, it has been observed that $v$ can be optimized at 3. Yet, test results with $v = \{0.5, 1, 2, 3\}$ are presented. For exhaustive search approach, a class-common modality selection is performed, which means a common selection is applied for all classes. It is expected that the exhaustive search gives the optimal selection, however our consideration is to see whether the class-specific approach of RELIEF-RDR can obtain better accuracy than the best possible class-common selection. Thus, exhaustive search requires calculating $2^6$ combinations of given 6 modalities. During comparison, not all of them, but the best selection is presented. Lastly, for generating RELIEF-F weights, we prefer a class-specific weight scheme by executing the algorithm once for each class, for a more detailed test. While using the RELIEF-F weights, the weights are used in two alternative ways: (i) The threshold for selection is preferred as 0 and the features with weights larger than 0 are combined, (ii) The interval $[-1, 1]$ of the weights are normalized into $[0, 1]$ and all of the weights are used.

In addition to these comparisons, the accuracies of RELIEF-F and RELIEF-RDR at different $k$ (number of nearest hit/miss selections) values are measured.

### 4.2 Results and Evaluation

Table 1 compares the best accuracies of single modalities, exhaustive search, RELIEF-F and RELIEF-RDR. The table presents the AP of all classes as well as the MAP values. Exhaustive Search result is given when the selection gives the best accuracy, which is the combination of Visual-Color, Visual-Texture and Audial-Cepstral modalities. Similarly, presented RELIEF-F and RELIEF-RDR results are the best accuracies obtained with optimal parameters. RELIEF-F is optimized at $k = 20$ nearest hits/misses. Besides, RELIEF-RDR is optimized at $k = 1200$, also with an additional parameter of $v = 3$.

The accuracy results on Table 1 show that combination of different modalities give better results than the single modalities. However, selection of modalities is a critical issue. A wrong selection can lead to worse results than the best of the single modalities. For instance, simple methods like averaging all modalities are affected by the unfavorable modalities and dependency among. In addition, it can be observed that the performances of single modalities vary in different classes. For instance; for "Animal" class "Visual-Texture" modality gives the best results, whereas "Textual" modality is better for "Flag-US" class. Such results validate the idea of exploiting class-specific features.

According to MAP values, we see that RELIEF-RDR ob-

tains better accuracy than the exhaustive search, that means class-specific selections of RELIEF-RDR are better than the best class-common selections. Besides, RELIEF-F cannot reach that boundary, although it has also generated class-specific selections. Besides, RELIEF-F and RELIEF-RDR are in compete, considering the APs across different classes. However, RELIEF-RDR is definitely superior in total.

An important discussion is the independency of modalities. Using complementary features with the methods requiring independent inputs can cause a decrease in the accuracies. In this study, the modalities utilized are not fully independent. RELIEF-F approach is known to be good at handling features with high dependencies. Also, exhaustive search can handle the dependency issue since it tries to find the optimal solution. The test results show that CSF approach is as successful as these two approaches at eliminating complementary features and selecting the most informative ones.

In addition to the general comparison given, a detailed analysis on the performances of RELIEF-F and RELIEF-RDR can be found in Figure 1. The graph presents the MAP values for different $k$ (number of nearest hit/miss selections) values. However, the used dataset is an unbalanced one, having different number of shots for each concept. So, when the number of shots is not enough to find $k$ nearest or hits, all of the hits/misses are used.

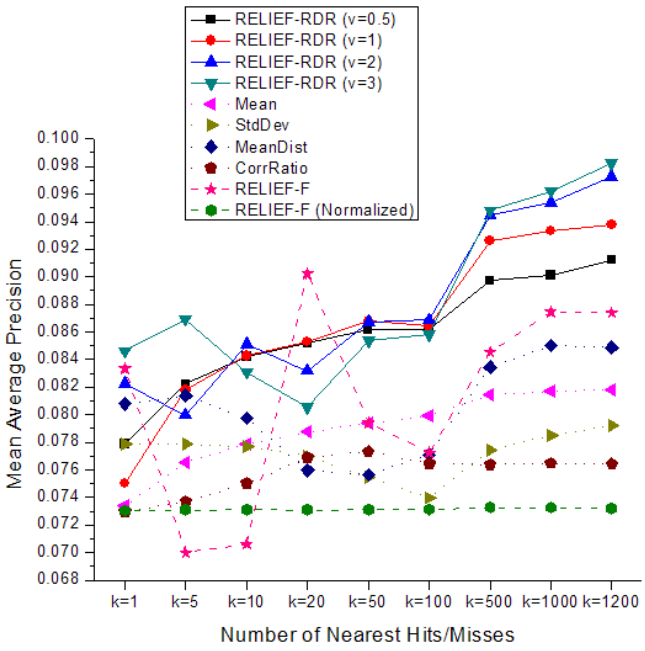


**Figure 1: RELIEF-F vs. RELIEF-RDR Comparison**

Considering Figure 1, it can argued that the accuracy of RELIEF-F highly depends on $k$ and the noise in the dataset. As explained in [17] and [27], it is expected to have an optimal value of $k$ which gives the best accuracy. In [27], it is advised to select $k = 10$ for better performance. In our test, the optimal point of k is observed as 20. However, the moves in the RELIEF-F accuracy graph are still different than the expected behavior, explained in Subsection 2.3. The major reasons why RELIEF-F have such drastic changes are the high noise and the unbalance of the dataset.

Besides, RELIEF-RDR is more robust against noise, and have an increasing tendency with the increasing $k$ values. Considering that RELIEF-F uses the instances one by one, the effect of noise is high. Alternatively, RELIEF-RDR first groups the instances, then use the statistics (average, standard deviation, etc.) to calculate the weights. So, the effect of the noise on the algorithm is less. In addition, RELIEF-RDR is not limited to the discrimination capability of features, it also benefits from the representation and reliability capabilities. The increasing trend of $Mean$, $StdDev$ and $CorrRatio$ in the the graph present such situation. Thus, RELIEF-RDR obtains better results than RELIEF-F for most of the $k$ values.

Yet, $MeanDist$, which is the discriminative parameter of RELIEF-RDR, is the major component in RELIEF-RDR. Among four parameters, it has best accuracy values. In addition, having higher powers of $MeanDist$ increases the accuracy (until some point), but makes the trend line of RELIEF-F more similar to RELIEF-F, by having more rises and falls. Thus, it can be argued that the discrimination capability of modalities are more affected from the noise than the representation and reliability characteristics. And also, $Mean$, $StdDev$ and $CorrRatio$ can be defined as the stabilizing parameters of RELIEF-RDR.

Besides, the normalized RELIEF-F weights give more stabilized results, too. However normalized version provides worse accuracies than the original weights. This is probably because of the fact that normalization of weights causes the the weights lose their influence on separability. So, the RELIEF-F weights should be used as what they are, by determining a threshold.

As a final remark, it can be asserted that the general view of the graph demonstrates the superiority of RELIEF-RDR against RELIEF-F.

## 5. CONCLUSION

In this paper, a new RELIEF extension, namely RELIEF-RDR, for utilization in multimodal information retrieval is proposed. RELIEF-RDR handles the inefficiencies of RELIEF on complex, multi-labeled and noisy features like multimedia data by introducing additional parameters in weight calculation. The algorithms benefits from the representation and reliability characteristics of features as well as the discrimination capability. In addition, the proposed algorithm extends the class-specific feature selection understanding of RELIEF by implicitly including it into the algorithm. The approach is tested on TRECVID 2007 dataset with visual, audial and textual features in a multimodal information fusion scenario. The results show that proposed algorithm is superior than RELIEF-F and class-common exhaustive search. Thus, it is argued that proposed approach is a timely efficient, accurate and robust way of modality selection.

## 6. ACKNOWLEDGMENTS

This work is supported in part by a research grant from TÜBİTAK EEEAG with grant number 109E014. The authors also would like to thank Nobuhiro Kaji for his valuable comments to improve the quality of the paper.

## 7. REFERENCES

[1] P. K. Atrey, M. S. Kankanhalli, and J. B. Oommen. Goal-oriented optimal subset selection of correlated

multimedia streams. *ACM Trans. Multimedia Comput. Commun. Appl.*, 3, February 2007.

[2] B.Mathieu, S.Essid, T.Fillon, J.Prado, and G.Richard. Yaafe, an easy to use and efficient audio feature extraction software, 2010. Proceedings of the 11th ISMIR conference, Utrecht, Netherlands.

[3] E. Bruno and S. Marchand-Maillet. Multimodal preference aggregation for multimedia information retrieval. *Journal of Multimedia*, 4(5):321–329, 2009.

[4] T. G. Dietterich. Machine-learning research: Four current directions. *The AI Magazine*, 18(4):97–136, 1998.

[5] F. J. Ferri, P. Pudil, M. Hatef, and J. Kittler. Comparative study of techniques for large-scale feature selection, 1994. Pattern Recognition in Practice IV, Multiple Paradigms, Comparative Studies and Hybrid Systems.

[6] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *J. Mach. Learn. Res.*, 3:1157–1182, Mar. 2003.

[7] M. A. Hall. *Correlation-based Feature Subset Selection for Machine Learning*. PhD thesis, Dept. of Computer Science, Univ. of Waikato, New Zealand, Apr. 1999.

[8] E. B. Hunt, P. J. Stone, and J. Marin. *Experiments in induction / Earl B. Hunt, Janet Marin, Philip J. Stone*. Academic Press, New York :, 1966.

[9] A. Jain, K. Nandakumar, and A. Ross. Score normalization in multimodal biometric systems. *Pattern Recognition*, 38(12):2270 – 2285, 2005.

[10] A. K. Jain, R. P. Duin, and J. Mao. Statistical pattern recognition: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:4–37, 2000.

[11] A. Jakulin and I. Bratko. Analyzing attribute dependencies. In N. Lavrac, D. Gamberger, H. Blockeel, and L. Todorovski, editors, *PKDD*, volume 2838 of *LNCS*, pages 229–240. Springer, 2003.

[12] M. Kankanhalli, J. Wang, and R. Jain. Experiential sampling on multiple data streams. *Multimedia, IEEE Transactions on*, 8(5):947 –955, 2006.

[13] K. Kira and L. A. Rendell. A practical approach to feature selection. In *Proc.of the 9th Int. Workshop on Machine Learning*, ML '92, pages 249–256, San Francisco, CA, USA, 1992. Morgan Kaufmann Publishers Inc.

[14] J. Kittler. Feature Set Search Algorithms. *Pattern Recognition and Signal Processing*, pages 41–60, 1978.

[15] J. Kludas, E. Bruno, and S. Marchand-Maillet. Information fusion in multimedia information retrieval. In *Proc. of 5th Int. Workshop on Adaptive Multimedia Retrieval (AMR)*, Paris, France, July 5-6 2007.

[16] J. Kludas, E. Bruno, and S. Marchand-Maillet. Can feature information interaction help for information fusion in multimedia problems? *Multimedia Tools Appl.*, 42:57–71, March 2009.

[17] I. Kononenko. Estimating attributes: analysis and extensions of relief. In *Proc.of European Conf. on Machine Learning*, pages 171–182, Secaucus, NJ, USA, 1994. Springer-Verlag New York, Inc.

[18] H. Liu, H. Motoda, and L. Yu. A selective sampling approach to active feature selection. *Artif. Intell.*, 159:49–74, November 2004.

[19] J. Martínez. Mpeg-7 overview (version 10). Requirements ISO/IEC JTC1 /SC29 /WG11 N6828, International Organisation For Standardisation, Oct 2003.

[20] P.Atrey, M.Hossain, A.E.Saddik, M.Kankanhalli. Multimodal fusion for multimedia analysis: a survey. *Multimedia Systems*, 16:345–379, 2010.

[21] MPEG. Mpeg-7 reference software experimentation model, 2003. http://standards.iso.org/ittf/PubliclyAvailableStandards /c035364_ISO_IEC_15938-6(E)_Reference_Software.zip.

[22] P. Over, G. Awad, W. Kraaij, and A. F. Smeaton. Trecvid 2007–overview. In *TRECVID'07*, 2007.

[23] B. B. Pineda-Bautista, J. A. Carrasco-Ochoa, and J. F. M. Trinidad. General framework for class-specific feature selection. *Expert Syst. Appl.*, 38(8):10018–10024, 2011.

[24] N. Poh and J. Kittler. *Multimodal Information Fusion: Theory and Applications for Human-Computer Interaction*, chapter 8, pages 153–169. Academic Press, 2010.

[25] J. R. Quinlan. Induction of decision trees. *Mach. Learn.*, 1:81–106, March 1986.

[26] M. Robnik-Sikonja and I. Kononenko. An adaptation of relief for attribute estimation in regression. In D. H. Fisher, editor, *ICML*, pages 296–304. Morgan Kaufmann, 1997.

[27] M. Robnik-Šikonja and I. Kononenko. Theoretical and empirical analysis of relieff and rrelieff. *Mach. Learn.*, 53:23–69, October 2003.

[28] Y. Saeys, I. n. Inza, and P. Larrañaga. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23:2507–2517, September 2007.

[29] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Commun. ACM*, 18:613–620, November 1975.

[30] M. R. Sikonja. Speeding up relief algorithm with k-d trees. In *Proceedings of Electrotehnical and Computer Science Conference (ERK'98)*, pages 137–140, 1998.

[31] L. Snidaro, R. Niu, G. Foresti, and P. Varshney. Quality-based fusion of multiple video sensors for video surveillance. *SMC-B: Cybernetics, IEEE Trans. on*, 37(4):1044 –1051, 2007.

[32] C. G. M. Snoek and M. Worring. Multimodal video indexing: A review of the state-of-the-art. *Multimedia Tools and Applications*, 25(1):5–35, 2005.

[33] Y. Sun. Iterative relief for feature weighting: Algorithms, theories, and applications. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(6):1035–1051, 2007.

[34] L. Wang, N. Zhou, and F. Chu. A general wrapper approach to selection of class-dependent features. *IEEE Transactions on Neural Networks*, 19(7):1267–1278, 2008.

[35] Y. Wu, E. Y. Chang, K. C.-C. Chang, and J. R. Smith. Optimal multimodal fusion for multimedia data analysis. In *Proc. of the 12th ACM Multimedia*, pages 572–579, New York, NY, USA, 2004. ACM.

[36] T. Yilmaz, A. Yazici, and Y. Yildirim. Exploiting class-specific features in multi-feature dissimilarity space for efficient querying of images. In *FQAS*, pages 149–161, 2011.