VISUAL CONCEPT DETECTION BY STACKED GENERALIZATION

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

MASHAR TEKIN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

JULY 2014

Approval of the thesis:

VISUAL CONCEPT DETECTION BY STACKED GENERALIZATION

submitted by **MASHAR TEKIN** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen _____
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Adnan Yazıcı _____
Head of Department, **Computer Engineering**

Prof. Dr. Fatoş Tünay Yarman Vural _____
Supervisor, **Computer Engineering Dept., METU**

Examining Committee Members:

Prof. Dr. Göktürk Üçoluk
Computer Engineering Dept., METU _____

Prof. Dr. Fatoş Tünay Yarman Vural
Computer Engineering Dept., METU _____

Assist. Prof. Dr. Sinan Kalkan
Computer Engineering Dept., METU _____

Dr. Ahmet Sayar
TÜBİTAK UZAY _____

Dr. Mete Özay
School of Computer Science, Univ. of Birmingham _____

Date: 24.07.2014

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: MASHAR TEKIN

Signature :

ABSTRACT

VISUAL CONCEPT DETECTION BY STACKED GENERALIZATION

Tekin, Mashar
M.S., Department of Computer Engineering
Supervisor : Prof. Dr. Fatoş Tünay Yarman Vural

July 2014, 60 pages

In this thesis, we propose a new Stacked Generalization method, called Fuzzy Stacked Generalized Ranking Optimizer, to optimize the ranking performances of visual concept detection systems. In the proposed method, fuzzy k-NN classifiers are employed in the base-layer. Then, a classifier selection algorithm is employed to select the classifiers which will be combined in meta-layer. Finally, the results of the selected classifiers are combined and classified by a fuzzy k-NN meta classifier. In the experiments, the proposed method performs better than the state of the art ensemble learning methods.

Keywords: Visual Concept Detection, Stack Generalization, Classifier Selection, fuzzy k-NN

ÖZ

YIĞIN GENELLEME İLE GÖRSEL KAVRAM TANIMA

Tekin, Mashar
Yüksek Lisans, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi : Prof. Dr. Fatoş Tünay Yarman Vural

Temmuz 2014 , 60 sayfa

Bu tezde, görsel kavram tanıma sistemlerinin sıralama performanslarını iyileştirmek için Bulanık Yığın Genelleme ile Sıralama İyileştirici isimli yeni bir Yığın Genelleme metodu önerilmektedir. Önerilen metotda, temel seviyede bulanık k-NN sınıflandırıcıları kullanılmıştır. Daha sonra, bir sınıflandırıcı seçme algoritması kullanılarak meta seviyede birleştirilecek olan sınıflandırıcılar seçilmiştir. Son olarak, seçilen sınıflandırıcılara ait sonuçlar birleştirilmiş ve bulanık k-NN meta sınıflandırıcısı tarafından sınıflandırılmıştır. Yapılan deneylerde önerilen yöntemin en güncel bütünleşik öğrenim metotlarından daha iyi çalıştığı gözlemlenmiştir.

Anahtar Kelimeler: Görsel Kavram Tanıma, Yığın Genelleme, Sınıflandırıcı Seçimi, fuzzy k-NN

To my father

Erdoğan Tekin

ACKNOWLEDGMENTS

I would like to thank my supervisor Professor Fatoş Tünay Yarman Vural for her constant support, guidance and friendship. It was a great honor to work with her.

I would like to also express my thanks for their assistance to Ersin Esen, Ahmet Saracoğlu, Medeni Soysal, Tuğrul Kaan Ateş, Ezgi Can Ozan and Kamil Berker Loğoğlu with their support, preparation of this thesis became much easier.

I would like to thank my friends in Data Processing Group of Space Technologies Research Institute for the great research environment they had provided. I have learned so much from their experience and suggestions.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ALGORITHMS

ALGORITHMS

CHAPTER 1

INTRODUCTION

Concept detection is the problem of detecting predefined concepts in a multimedia document database such as image, audio or video. The explosive and rapid growth in the amount of multimedia content on the television and Internet services like YouTube and Facebook brings a demand for techniques to easily search, filter and organize the huge amount of data. Managing broadcast archieve of a broadcasting agency or retrieving videos related with a concept from digital libraries in an effective way, are some of the problems waiting to be solved in this area. Since it is not feasible to manually annotate all videos, visual concept detection on videos becomes a significant research topic in the computer vision community.

The major problem of visual concept detection is the *semantic gap* which is defined by Snoek et al. [53] as: "The lack of correspondence between the low-level features that machines extract from video and the high-level conceptual interpretations a human gives to the data in a given situation." Generally speaking, concept detection systems aim at bridging the semantic gap between the low level features and high level semantic concepts. This is a very difficult task in a concept detection system, basically because of the fact that a semantic concept may have diverse representation in terms of low level color, texture and shape features.

In order to bridge the semantic gap, most of the concept detection systems extract visual and/or audio features from video and classify these features into predefined concepts by using machine learning techniques. In these systems,

various feature types and classifiers are successfully employed to utilize different aspects of concepts for classification. Most of these systems fuse classifiers using a Stack Generalization (SG) architecture which is an an ensemble learning technique and aims at optimizing the system performance by combining multiple classifiers under a hierarchical structure. Generally speaking, small number of base-layer classifiers are employed in SG architectures. As a result, faster training time and better generalisation performance are achieved than using large number of classifiers. These systems are successful for the detection of small number of concepts. However, the performances of these systems decrease when the number of detected concepts is increased. This is basically because of the fact that the variety of features related to individual classifiers is not enough to effectively represent the large number of concepts. For this purpose, methods which employ a large number of features and classifiers are proposed. The advantage of these systems is that they employ one or more classifiers to effectively detect most of the concepts. However, for the detection of each concept either all classifiers or a subset of classifiers are selected and combined by the users. This approach may degrade the performance of the systems, named as *black art* problem, since the independency and representation power of features related to classifiers are not considered while determining the classifiers to combine. The relationship among the classifiers which effects the performance is called the *black art* problem.

In this thesis, we propose a new SG architecture, named as *Fuzzy Stacked Generalized Ranking Optimizer (FSG-RO)*, which employs a classifier selection method in order to resolve the black art problem and optimize the performance of concept detection systems. In FSG-RO, a subset of classifiers related to features are selected from the classifiers employed in the system by using a classifier selection method and selected classifiers are employed for the detection of a concept. The advantage of FSG-RO is that, mostly, it discovers the subset of classifiers which optimize the performance of the system when compared to the individual classifiers performances and the performance of the system that uses all classifiers. On the other hand, for the cases that individual classifier performances are poor FSG-RO can not optimize the system performance.

The proposed FSG-RO architecture is used in a visual concept detection system on broadcast media. The system is designed as a ranking-based application where the results of the detection process of a concept is a ranking list of samples according to their membership values for the given concept. Various local, global and key-point based features, such as color and texture features are employed in the system. FSG-RO architecture, which uses fuzzy k-NN classifiers in its hierarchical levels (base-layer and meta-layer), is used to combine the classifiers and optimize the ranking results of the system.

Unlike the common case for visual concept detection systems, using a classifier selection method to optimize the ranking results is the key contribution of our study. We implement a classifier selection method to find the set of classifier which will boost the performance of the system.

This thesis is organized in six chapters. In Chapter 2, the overview of the concept detection systems for the image, audio and video domains is given and the techniques used in the concept detection systems are briefly explained. In Chapter 3, the stacked generalization methods in the literature are reviewed since these methods are commonly used in concept detection systems. In Chapter 4, the details of the proposed FSG-RO architecture and FSG architecture are given. In Chapter 5, the experimental results and comments about these results are presented. Finally in Chapter 6, the summary of our study and some remarks for future works are given.

CHAPTER 2

AN OVERVIEW OF CONCEPT DETECTION SYSTEMS

Semantic concept is defined by Snoek et al. [53] as: "An objective linguistic description of an observable entity." Based on this definition, the difference of the concepts from object categories is that the events, like meeting or fire are also defined as concept. In recent years, in order to effectively search and index multimedia documents concept detection systems are studied.

In this chapter, an overview of the studies related to concept detection is presented. Since concept detection is a popular area for researchers, studies are available on different domains, such as image, audio and video. In order to present the available systems in a comprehensive way, we present our overview in two sections. In the first section, concept detection systems on image, audio and video domains are demonstrated. In the second section, a review of the strategies used for concept detection systems is provided.

2.1 Concept Detection Systems

Image, audio and video are the main domains that concept detection systems are used. For each of these domains, an overview of the available systems are given in the following sections.

2.1.1 Image Concept Detection Systems

In image domain, many expert systems for different concepts are studied. A popular example is proposed by Viola and Jones [69], where a face detection method with different profile views is developed. Also, Dalal and Trigs [16] propose a successful method for human detection in images. In other studies, plant identification, medical image retrieval and robot vision tasks are accomplished for ImageCLEF, which is an evaluation benchmark [10]. Szummer and Picard [60] focus on the classification of Indoor/Outdoor images. Since, the methodologies and features used in the expert systems are specifically designed for a concept, these systems are successfully used in their own application domains, but can not be used for any other concept detection systems.

Due to the time and computational complexity, it is not feasible to design an expert system for each concept. As a result, large scale concept detection systems that detect large number of concepts in a generic framework are studied. In ImageCLEF [32], successful solutions for large scale visual concept detection are presented. Tahir et al. [61] and Sande et al. [67] propose frameworks for detection of 53 concepts, such as, animals, vehicles, and plants, on consumer photos using only low level visual features. In the study of Huiskes et al. [24] a framework using social data such as Flickr tags together with low level visual features is proposed to achieve significant enhancements on the performance of image retrieval. Since, same framework and features are used for the detection of each concept, the performance of large scale concept detection systems are worse than the performance of expert systems. These systems should be studied in more detail in order to common usage in real life applications like image based search engines.

2.1.2 Audio Concept Detection Systems

Audio based semantic classification systems are developed on isolated data, which contains only pre-determined sounds, such as sound effects in audio databases [29, 38], broadcast data that contains mixed audio of various sounds like in tele-

vision broadcast [34, 37, 39] or surveillance data that are the sounds obtained from surveillance cameras [5, 14].

The framework demonstrated in the study of Pfeiffer et al. [38] is one of the first work for audio content analysis. This study is about music indexing and violence detection in sound tracks from isolated dataset. Violence detection is labeled as gunshots, cry and explosion sounds. In the study of Mesaros et al. [29], there are 61 audio classes for classification, where isolated sound effects of real life audio are used. In the study of Portelo et al. [39] detection of 15 different non-speech events, such as, jet sounds, vehicle sounds, water sounds on movies are detected. A system proposed by Petridis et al. [37] detects the speech and 5 non-speech audio events, namely music, sound of water, sound of air, engine sounds and applause, on News Broadcasts. Ozan et al. [34] classify 17 different audio event on TV broadcast. Clavel et al.[14] propose a surveillance event detection system which detects the abnormal audio events especially for the shot detection in noisy environments like public places. Another surveillance event detection system proposed by Atrey et al. [5] classifies events like, footstep, door knock, talk and shout.

In audio domain, systems used to classify isolated data achieves high performance and can be used for audio databases. Additionally, the classification of speech and music is also achieved successfully. However, in order to effectively use audio retrieval systems in real life applications, the performances of the systems that classify audio events in broadcast data need to more optimized. Also, the number of the events should be increased.

2.1.3 Video Concept Detection Systems

In the video domain, a rapid increase has been observed in researches in recent years. Using audio and/or visual clues variety of systems were proposed by Smeaton et al. [47] and Wang et al.[72]. Systems for the semantic classification of small number of concepts up to hundreds of concepts were demonstrated [20, 23, 49, 50]. The advancements in semantic analysis of videos can be observed in TRECVID, an annual evaluation benchmark for research groups. In TRECVID

2006 [33], 20 concepts has been selected for semantic analysis, while in 2010 and in 2011 this number is increased up to 130 and 346, respectively.

The study of MediaMill group [49, 50] is one of the most prominent studies in the literature. In [49] they classified a lexicon which contains 32 specific and general concepts in one system architecture on broadcast news data. In [50] they extended their framework to classify 130 concepts, such as, classroom, door, cityscape and singing etc. Although, the system of MediaMill achieved one of the best performances on the average in TRECVID 2010, for many concepts the performance of the system is far from being useful. The variety and number of features employed in their system is not enough to achieve good performances for all concepts.

For consumer videos, Chang et al. [20] propose a large scale concept detection system a pioneer work classifies 25 different concepts which were determined for the needs of consumers. Some of the concepts classified are graduation, park, baby and cheering. In their study, visual features and audio features are extracted from video and Support Vector Machine (SVM) [8] is used for classification of each feature. Then, the independent classification scores are combined by using various weighted sum strategies. The system achieves prominent performances for some of the concepts, however there are significant number of concepts such as picnic, animal and baby that can not be successfully detected by the system. The weaknesses of this study is that the visual and audio features are not enough to effectively describe each concept and the combination methods are simple.

In the study of Soysal et al. [56], KavTan multi-modal concept detection system is proposed for the detection of 28 different concepts on broadcast media. In the study, in order to increase the performance of the system specialized methods are employed for the detection of concepts such as Nudity, Blood and Human Presence. On the other hand, most of the visual concepts are detected by a generalized visual concept detection module which employs an ensemble learning architecture Stacked Generalization(SG) [74]. In the module, large number of visual features consists of color, texture and keypoint based features are extracted

and SVM classifiers are used for the classification of features. Then, the predictions of the classifiers are concatenated and classified by another SVM to get final prediction. Employing large variety of features and using SG architecture are the superiorities of the KavTan system when compared to the previous studies. However, the weakness of this system is that for the detection of each concept either all features or the features selected by users are employed together. This may decrease the performance of the system, named as *black art* problem [74] in the literature. In order to solve this problem, some feature/classifier selection methods can be used.

In order to design a concept based video retrieval system, which achieves performances comparable to text retrieval systems on the web, available video concept detection systems need further improvements. Hauptmann et al. [23] state that the number of concepts detected by these systems should be increased up to 5000. Also, the performances of these systems should be optimized by increasing the variety of features employed in the systems, using feature selection methods and more sophisticated ensemble learning techniques.

2.2 Concept Detection Methodologies

As mentioned in the previous section, in order to achieve the semantic analysis problem of multimedia data, various systems are proposed [53]. Since the problem has different aspects and sub-tasks such as feature extraction and classification, many algorithms and state-of-the-art methodologies are used in these system to achieve better performances. A review of methodologies for different domains are provided in [53, 57, 72]. In this section, we will review the methodologies used in video domain since the scope of our thesis is visual concept detection on video.

In recent years, generic systems which employ a single framework to detect large number of concepts are proposed for video concept detection. The basic framework of these systems contains three basic steps, namely, video segmentation, feature extraction and machine learning. In the video segmentation step, video

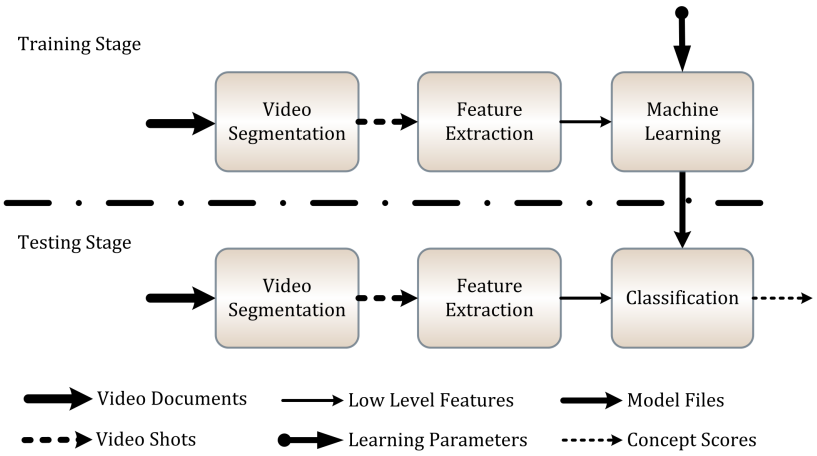

Figure 2.1: Basic framework for a generic concept detection system.

is partitioned into segments. Then, in the feature extraction step low level features are extracted from these segments. Finally, machine learning techniques are employed to classify these features. In order to employ machine learning techniques, training and testing stages are executed in basic framework. In Figure 2.1 steps of the basic framework are shown for training and testing stages.

In order to improve the performance of generic systems, researchers extend their frameworks using supplementary steps, namely, feature fusion, classifier fusion and modelling relations. One or more of these steps may be employed by frameworks. Feature fusion step is used to combine the various features extracted from video segments in order to obtain a single feature vector for classification. On the other hand, classifier fusion is used to combine the results of various classifiers to obtain a single result. Finally, modelling relations are used to optimize the detection results by using the relations between concepts. The general framework using supplementary steps is shown in Figure 2.2 for testing stage.

In order to achieve each of the basic and supplementary steps, many techniques are used by available systems [53], such as, feature/classifier fusion and modelling relations. A tree of the techniques used for each step is shown in Figure 2.3. In following sections, the explanation of these techniques are given for the steps *Video Segmentation*, *Feature Extraction*, *Feature Fusion*, *Machine Learning*, *Classifier Fusion* and *Modelling Relations*.

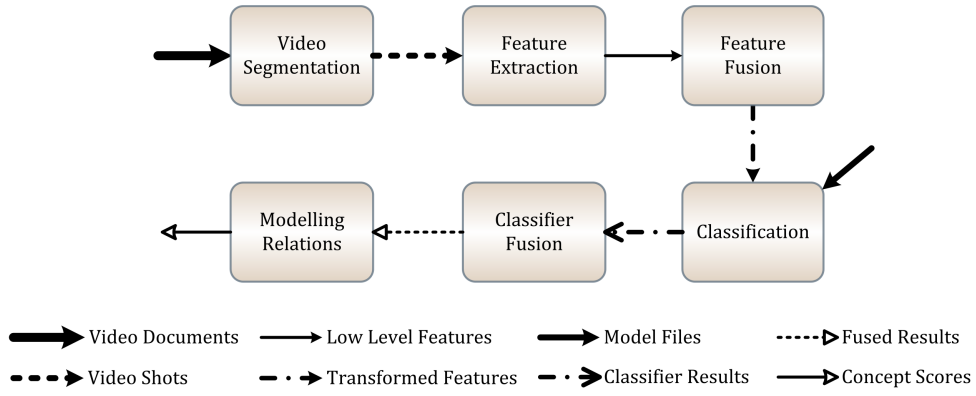

Figure 2.2: General framework for a generic concept detection system.

2.2.1 Video Segmentation

In order to design a video concept retrieval system at a fine granularity, videos are partitioned into workable segments. The most convenient segments for concept detection are video shots, which are the sequences of the frames of a continuous camera action with respect to space and time [78]. *Video Shot Detection*, which is defined as automatically detecting the shot boundaries, achieved by measuring the amount of the change between successive frames of a video. If the amount of change exceeds a threshold, then a shot boundary is detected between these frames [79].

In order to analyse the video shots and decrease the computational complexity, a video shot is generally represented by one or more frames, named as *keyframes*. *Keyframe Extraction* is often achieved by taking the central frame of a video shot as *keyframe*. In the study of Soysal et al. [56], *keyframes* are obtained by uniformly sampling the frames throughout the shot duration.

Video shot detection and keyframe extraction processes are successfully achieved by concept detection systems in the literature [56, 79]. In feature extraction step, low level features are extracted from the shots and keyframes obtained in this step.

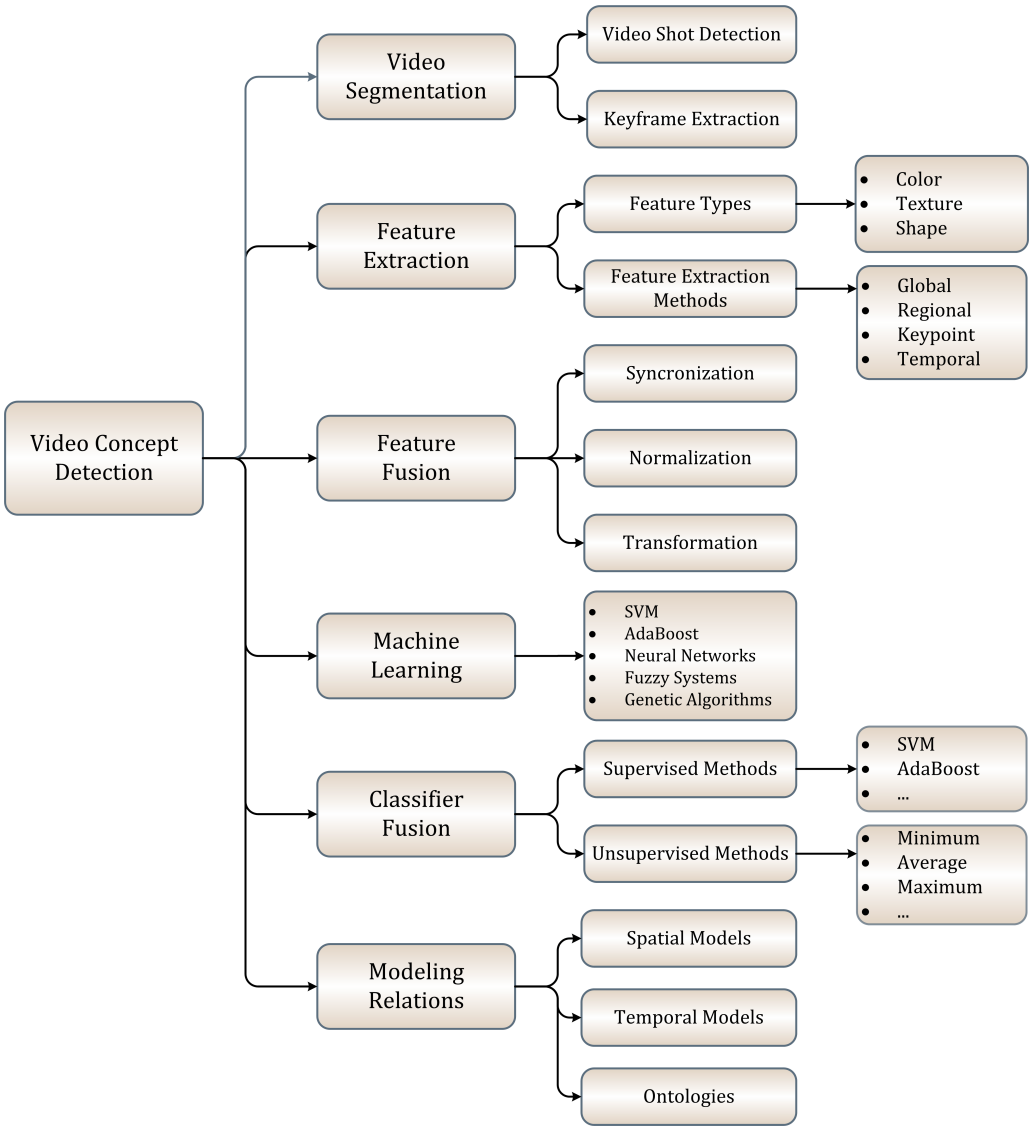

Figure 2.3: A hierarchical scheme, representing the methodologies used by concept detection systems.

2.2.2 Feature Extraction

In feature extraction step, the popular *Global* [59], *Regional* [30], *Keypoint-based* [68] and *Temporal* [12, 68] feature extraction methods used. In the *Global* method, features are extracted from the entire keyframe in order to obtain information about the whole [59]. When the global features are insufficient for the analysis of a keyframe, *Regional* feature extraction methods are used. In this method, generally, keyframe is partitioned into regions with fixed set of rectangles rely on different locations of image [30] and features are extracted from these predefined rectangles. In *Keypoint* scale, features are extracted from the points sampled from a keyframe [68]. *Keypoint based* features come into prominence in many concept detection systems [16, 33] with the proposition of *Scale-Invariant Feature Transform (SIFT)* by Lowe [27]. *Temporal* features are only extracted from videos using temporally ordered images belong to video shots and used to describe motion and activities in a video shot. Motion Activity Descriptor, Camera Motion Descriptor are temporal features commonly used in the literature [12, 68].

By using one or more of the presented feature extraction methods various feature types, namely, *color*, *texture* and *shape* features are extracted from keyframes in available concept detection systems. *Color* features are mostly extracted from different color spaces such as RGB, HSV space or the invariant sets of color spaces [12, 21] in order to describe the color characteristics of a keyframes. *Texture* features are used to classify various patterns such as pattern of grass, pattern of sand by capturing the similarity in local patterns [12, 44] and *shape* features are extracted from the regions that image is segmented into. Region Shape and Contour Shape are some of the commonly used descriptors defined by the MPEG-7 standard [12].

In our study, we use several color and texture features which are summarized as follows

- **Color Layout [43]:** Describes the spatial distribution of colors by using Discrete Cosine Transform.

- **Color Moments [58]:** First three moments of colors on 5x5 grid are computed.
- **Color Structure [43]:** Represents the spatial structure of the colors and their frequencies.
- **Co-occurrence Texture [22]:** Describes the entropy, energy and homogeneity of texture.
- **Dominant Color [43]:** Respresents the dominant colors and their statistics such as variance.
- **Edge Histogram [43]:** Describes the spatial distribution of edges for 16 image sub-regions.
- **Homogeneous Texture [43]:** Represents the texture of regions by using energy deviation and mean energy from a set of frequency channels modelled by Gabor functions.
- **Scalable Color [43]:** Color histogram which is calculated for HSV color space. Haar transform is used for encoding.
- **SIFT [27]:** Describe local image patches by using location, scale and rotation invariant feature vectors.
- **Wavelet Texture [75]:** By using a 3x3 grid, haar distributions among 12 sub-bands are obtained.

2.2.3 Feature Fusion

In most cases, a single feature is not enough for systems which aim at detecting many concepts since there may be more than one concept having the same characteristics for a single feature. For example, color characteristic of *sun* and *fire* may be same. As a result, large number of features from various feature types, such as, color, texture and shape, are used in concept detection systems. In order to obtain a single feature descriptor, these features are combined by using fusion techniques.

A key question in feature fusion is that which features are to be selected for fusion. In order to effectively use feature fusion, there should be some form of independence among the features [53]. There are two approach used to provide independency between features. One approach is using features from different modalities [73] and other approach is using different feature types extracted from same modality in such a way that features complete the defficiency of each other. In the study of Tseng et al. [65], various features from the same modality are fused by using the concatenation operation. Amir et al. [3] fused global and regional features.

Since the features extracted from different modalities, may have different units like frame or video shot, a *Synchronization* is needed. In the study of Snoek et al. [52] this problem is solved by selecting the common unit for features. Another problem is that, different feature types may have different dynamical ranges. This problem is solved by *Normalization* techniques in the literature, like in the study of Wilkins et al. [73]. The last and the most important problem for feature fusion is the curse of dimensionality problem of the machine learning techniques. In order to solve this problem, various feature transformation methods are used in the literature. Chen and Hauptmann [77] reduce the dimensionality of used features using Fisher's linear discriminant, and then concatenate the projected features. Soysal et al. [56] use *Bag of Visual Words (BoVW)* [76] approach to reduce the dimensionality. The set of feature descriptors extracted for a video shot, is projected to visual codeword histogram by using the visual codebooks obtained in the training phase using k-means clustering method.

2.2.4 Machine Learning

Since detecting a concept requires too many decision rules, machine learning techniques are used to learn the concepts. Simply, the aim of machine learning is to obtain a model, which achieves optimal generalization performance, by using a limited amount of training samples in order to optimize the classification performance of a concept.

In video concept detection systems, machine learning step is generally achieved

by using a supervised learning method where the classifier is trained by examples which are labelled previously. Training and testing stages employed using machine learning are shown in Figure 2.1. In training stage, the best possible configuration of features is learned using the training samples by classifiers. In testing stage, a classifier is used to make a decision for each feature vector for every concept.

The curse of dimensionality problem and optimizing the parameters of a classifier too intensively for the training samples are some of the reasons which decrease the performance of a classifier [53]. As a result, determining the number of features to employ and optimizing the classifier parameters should be carefully achieved by supervised learning methods. Moreover, learning from a limited number of training samples, handling data imbalance problem for training samples are other tasks that should be handled by classifiers. In order to achieve these tasks, Support Vector Machine (SVM) [8] is successfully used become a default choice for most of the concept detection systems [4, 6, 9, 20, 31, 42, 46, 49, 50, 56, 71].

In literature, the practical and fast k-NN method is employed in several concept detection systems [6, 42]. Although, it generally achieves a lower classification rates than SVM, this method is faster for training. Additionally, for the cases where large number of training samples are available, it may outperform the other classification methods, such as SVM and Neural Networks.

2.2.5 Classifier Fusion

Fusion of classification results is another approach used to optimize the performance of video concept detection systems. In the study of Snoek et al. [55] classifier fusion approach is experimentally compared to the feature fusion approach and it is shown that classifier fusion gives slightly better performance than feature fusion for most of the concepts. As a result, classifier fusion is preferred by many systems [4, 6, 9, 20, 31, 42, 46, 56].

In order to obtain a better performances than the performances of a single

classifier, there should be some independency among the fused classifiers. There are three methods in the literature in order to achieve independency between classifiers. The first and most used method is that separate features such as color, texture and motion features are used to train each classifier [4, 9, 20, 31, 46, 56]. In the second method, separate classifiers are trained for a feature type. Independency of classifiers can be achieved by using different machine learning techniques which may cover varying regions in feature space. In the studies of Ballas et al. [6] and Safadi et al. [42], k-NN and SVM classifiers are fused to optimize classification performance. Additionally, we can use same classification algorithm but vary the parameters of classifiers to obtain separate classifiers [70, 20]. In the study of Chang et al. [20] for each feature 25 SVM classifiers trained by using different parameters are fused. Wang et al. [70] fused the classification results of SVM classifiers with different kernel types. In the third and final method, classifiers are trained by using separate training data. In the study of Snoek et al. [54], new train datasets are obtained by re-sampling the original train data and for each train dataset a SVM is trained. Totally, 200 classifier is trained and combined to optimize the performance of the proposed system. All explained methods are successfully used to optimize the performance of concept detection systems, however, using various classifiers for a feature as in the second and third methods, increase the computational complexity of systems and rarely used in literature. On the other hand, using separate features is the common choice for concept detection systems.

Once the classifiers are obtained, the next step is to combine them. Among many methodologies, the popular methods for this task is combining classifier rankings, binary classification results or probabilistic decisions. Between these choices, almost all concept detection systems prefer to combine the probabilistic decisions of classifiers [4, 6, 9, 20, 31, 42, 46, 56]. In most cases, it gives better performances.

In order to combine the classification results *supervised* and *unsupervised* combination methods are used in the literature. In supervised methods, concept probabilities are concatenated using vector concatenation and classified by a classifier which is generally SVM [55, 77, 56]. In unsupervised methods, simple

functions are used to combine classifier results. Taking the average, minimum, maximum or geometric mean of the probabilities are some of these functions [26, 54, 51]. Because of the simplicity, unsupervised methods are commonly used by concept detection systems [4, 6, 20, 31, 46].

2.2.6 Modelling Relations

For a system which detects more than one concept simultaneously, the semantic relationship between concepts can be used to improve the results. For example, the presence of a boat together with water might increase the possibility of presence of these two concepts while decreasing the probability desert. In order to increase performance by using semantic relationships, the co-occurrence relations of the concepts should be exploited. For this purpose, two models are defined in the literature which are *Spatial* and *Temporal* models.

In *Spatial* models, concepts that are simultaneously available in the same frame are used to optimize detection results. For example, the presence of sand and water in a single frame, increases the probability of beach. This model is used in [25, 40].

In *Temporal* models, occurrences of the concepts in consecutive video frames or video shots are used. For example, the detection of the concept *Air-plane take-off*, can be achieved by the detection of *Air-plane* and *Sky* in successive video frames or video shots [18, 40]. In order to model dynamic concepts, ontologies are proposed for specific domains like soccer and medical [7, 19]. Both of the *Spatial* and *Temporal* models are successfully used for the detection of various concepts in the literature. Temporal models are used for *dynamic concepts* and Spatial models for others.

CHAPTER 3

AN OVERVIEW OF THE STACKED GENERALIZATION METHODS

Stacked Generalization (SG), introduced by Wolpert [74], is one of the general methods used in the literature for ensemble learning [13, 15, 17, 28, 62, 64, 66]. It is based on combining the predictions of several classifiers, as base-layer classifiers, in various ways in order to achieve better performance than the best individual base-layer classifiers. In the following section, firstly the SG architecture will be explained and then in subsequent section the SG architectures and various properties of this method will be discussed.

3.1 The Stacked Generalization Method

Stacked Generalization (SG) is a general method of using a high-level model to combine lower-level models to improve the overall predictive accuracy. The general structure of SG method is shown in Figure 3.1. High level model, called meta layer classifier, used to learn from the output of lower-level models, called base-layer classifiers.

Different type of base-layer classifiers may be used in SG methods. If the type of all base-layer classifiers are the same, such as all classifiers are SVM, it is named as *homogeneous base-layer classifiers*. On the other hand, if different type of classifiers are employed together in base-layers, such as, fuzzy k-NN and SVM, it is named as *heterogeneous base-layer classifiers*.

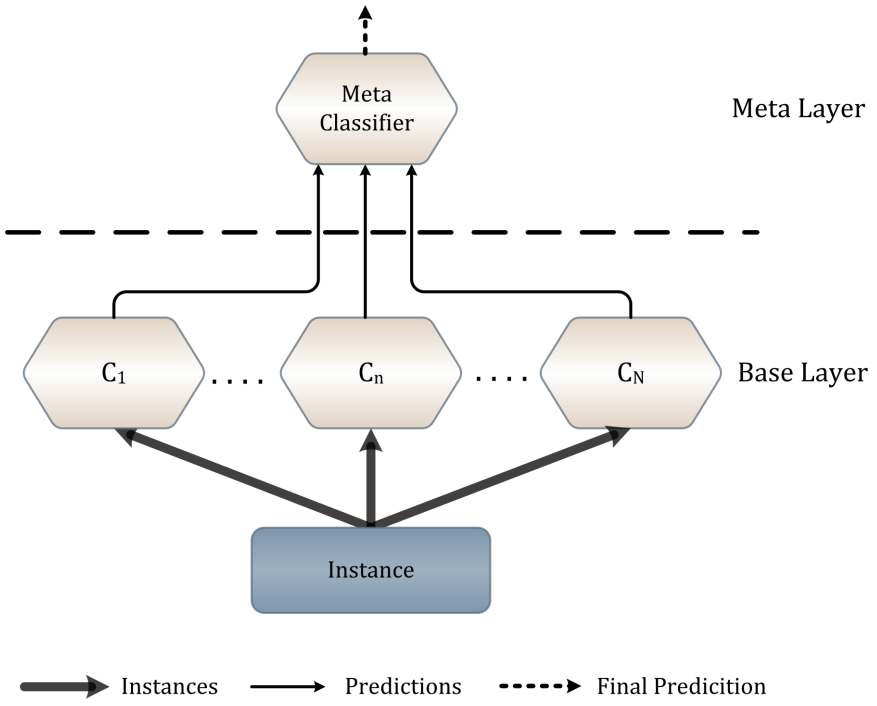

Figure 3.1: General structure of Stacked Generalization

3.2 Stacked Generalization Methods

Among a wide variety of SG methods in the literature, we review the methods similar to the proposed FSG-RO architecture where the decisions of the base-layer classifiers are fused by the linear combination or vector concatenation operation.

In the study of Sen and Erdogan [15] 13 different types of base level classifier such as k-NN, SVM with different kernel types, and binary decision tree etc. are employed. The combination of the classifiers is achieved by using a group sparse regularization method, a linear combination algorithm at the meta-layer. In our visual concept detection system we will employ homogeneous SG framework, because of the fact that using various type of classifiers in base-layer increase the computational complexity of a system. Therefor, in the rest of this section, homegeneous SG algorithms are reviewed.

In the study of Ueda [66], a prominent SG framework employing homegenous

base-layer classifiers is proposed. In the proposed framework, Neural Network classifiers are employed as the base-layer classifiers. In meta layer, each of the base-layer classifiers which achieves best classification performance for each class are selected to combine. This method aims at exploiting the strenghts of the base-layer classifiers. The selected classifiers are combined by using a linear combination method. Similarly, in the SG method of Ahmad and Zhang [1], Neural Networks are used in both layers and the classifiers to combine in meta-layer are selected using backward elimination and forward selection algorithms. Their system is tested on the database of diabetes. Using homogeneous base-layer classifiers and effective classifier selection methods are the superiorities of these methods. On the other hand, as mentioned in section 2.2.4 SVM and k-NN classifiers are more suitable for video concept detection systems and Neural Networks are not employed in recent video concept detection systems.

In the literature, prominent SG algorithms which employ fuzzy k-NN and SVM classifiers are also proposed. Shiraishi et al. [45] propose a multi-class classification system using SG architecture with SVM binary classifiers in base layer. In the study of Soysal et al. [56] SG framework is used in the visual concept detection module of a multi modal concept detection system (KavTan) for broadcast media. They use SVM classifiers for each type of visual feature, extracted from the video in base-layer and combine the probabilistic decisions of classifiers using vector concatenation operation. At meta-layer combined values are classified by another SVM producing probabilistic decisions. Employing large number of feature and using SVM classifiers are the superiorities of the KavTan system. On the other hand, the performance of KavTan system can be further improved by employing feature and/or classifier selection algorithms.

In the image annotation system of Akbas and Yarman Vural [2] fuzzy k-NN classifiers are employed in the layers and vector concatenation is used as the combining method. Similarly, Ozay and Yarman Vural [35] use SG architecture with fuzzy k-NN base-layer classifiers for their multi-class image classification system. At the base-layer, fuzzy k-NN classifiers are used to classify a set of feature vectors extracted from an image and each classifier outputs a membership value vector. In meta layer, these outputs are concatenated to build a linear

regression equation. In a recent study of Ozay and Yarman Vural [36], fuzzy k-NN classifiers are employed in the layers of the SG architecture which is named as Fuzzy Stacked Generalization (FSG). By using fuzzy k-NN classifiers, prominent performances are achieved by these systems. However, these architectures employ small number of features in the base layer and features are determined by the users. The performance of these architectures can be further improved by emloying a feature selection algorithm to determine the features used in the base-layer.

In the literatur, SG methods are successfully used to boost the performances of individual base layer classifiers. However, there are situations that the performance of the system decreases at the meta-layer. Employing classifiers which do not provide complementary information for the generalization accuracy is one of the reasons of this performance decrease. For SG method, identifying the correlation between performance of the system and different parameters of the methods is defined as "black art" problem in the studies of Wolpert, Ting and Witten [63, 74]. In order to solve this problem, a feature selection method can be employed to select the features whose combination improves the performance. In this thesis, we propose a new SG framework which employs a feature selection method and uses fuzzy k-NN classifiers in layers.

CHAPTER 4

FUZYY STACKED GENERALIZED RANKING OPTIMIZER (FSG-RO)

The major problem of visual concept detection is the *semantic gap* which is defined by Snoek et al. [53] as: "The lack of correspondence between the low-level features that machines extract from video and the high-level conceptual interpretations a human gives to the data in a given situation." Generally speaking, concept detection systems aim at bridging the semantic gap by extracting low level features from video frames or shots and classifying these features into predefined concepts by using machine learning techniques.

Bridging the semantic gap is a difficult task since a semantic concept may be appear with various shapes, colors and textures in a video. In order to detect a concept various features which describe different characteristics of a video are employed in concept detection systems. Most of these systems, use classifier fusion techniques (explained in section 2.2.5) to combine the classification results of various features.

In the concept detection systems, Stack Generalization (SG) is one of the methods used to employ various features and classifiers together for the detection of large number of concepts. However, most of these systems can not effectively use the different aspects of the concepts, since small number of pre-determined features are employed to classify composite concepts, such as fire and violence. On the other hand, the systems employing large number of features, generally, do not utilize a classifier selection method and utilize all base layer classifiers to detect all concepts. As a result, performance improvements that can be obtained

by using various classifier combinations for each concept, can not be explored.

In this thesis, we propose a concept detection system which employs a new SG architecture called *Fuzyy Stacked Generalized Ranking Optimizer (FSG-RO)*. In the FSG-RO architecture, at the base layer fuzzy k-NN classifiers which make fuzzy decisions are employed. At the meta layer, features are selected according to their ranking performances and the classification results of selected ones are combined using vector concatenation. Then, the generated vector is classified by a meta classifier, which is also a fuzzy k-NN classifier. The final output is also a fuzzy decision (membership value) which shows the probability of the presence of a concept. FSG-RO architecture produces a rank list and designed to optimize ranking performance of the system.

FSG-RO is based on the *Fuzzy Stacked Generalization (FSG)* architecture of Ozay and Yarman Vural [36]. Similar to FSG, FSG-RO employs fuzzy k-NN classifiers in layers. The major difference of FSG-RO is that a feature selection is included in the architecture. A brief explanation of FSG is given in following section and in subsequent section FSG-RO architecture is explained in detail.

4.1 FSG System Overview

Fuzzy Stacked Generalization (FSG) architecture, suggested by Ozay and Yarman Vural [36], is a two layered SG method which employs fuzzy k-NN classifiers in its layers. At the base-layer of FSG, fuzzy k-NN classifiers are employed to classify features extracted from samples. For a given sample, each individual classifier produces a class membership value vector and in data fusion stage these vectors are combined by using vector concatenation. Finally, at the meta-layer, the output of data fusion stage is classified using another fuzzy k-NN classifier and the class with the highest membership value is predicted as the class label of test sample. The general scheme of FSG architecture is shown in Figure 4.1.

In the study of Ozay and Yarman Vural, FSG is employed in order to optimize the accuracy of various multi-class classification systems that classify 10, 15 and 20 classes on image domain. In the experiments of FSG, it is shown that

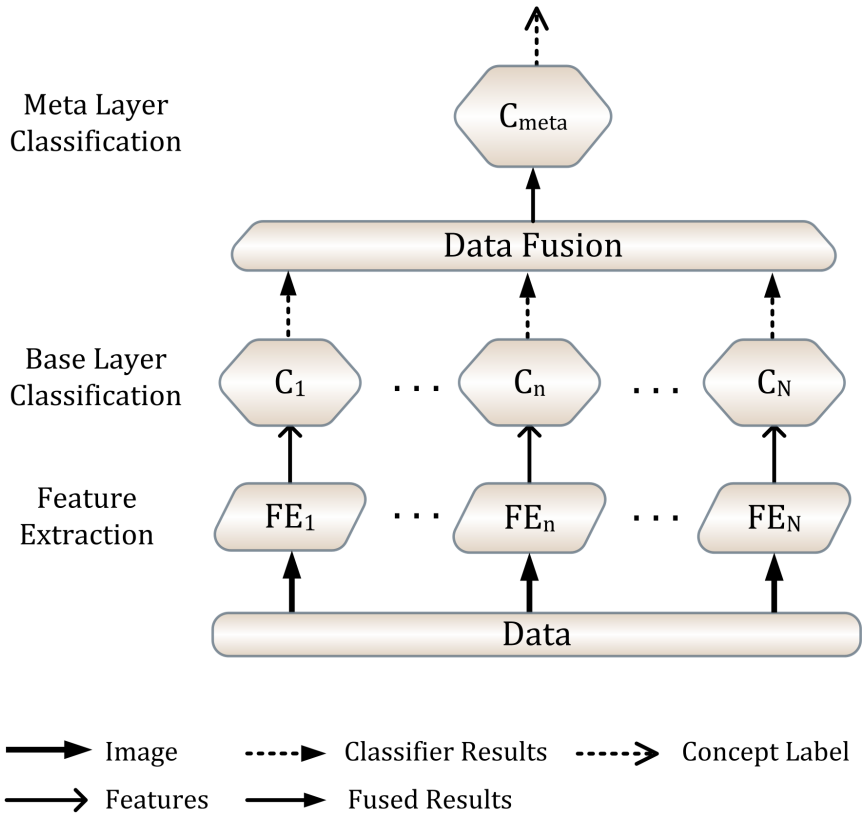

Figure 4.1: General scheme of FSG Architecture.

FSG boost the performance of individual base layer classifiers. Also, it achieves better performances than the state-of-the-art ensemble learning techniques such as Adaboost and Rotation Forest.

4.2 FSG-RO System Overview

FSG-RO architecture is designed to optimize the ranking performance of a generic video concept detection system which is implemented as a binary classification system. As a result, for each concept a separate FSG-RO system is trained as a dichotomizer and for a set of input samples, FSG-RO produces a rank list which contains the membership value of each sample for the concept.

The general structure of FSG-RO architecture is shown in Figure 4.2. Basically, FGS-RO consists of 5 modules, namely *Feature Extraction*, *Base Layer Classi-*

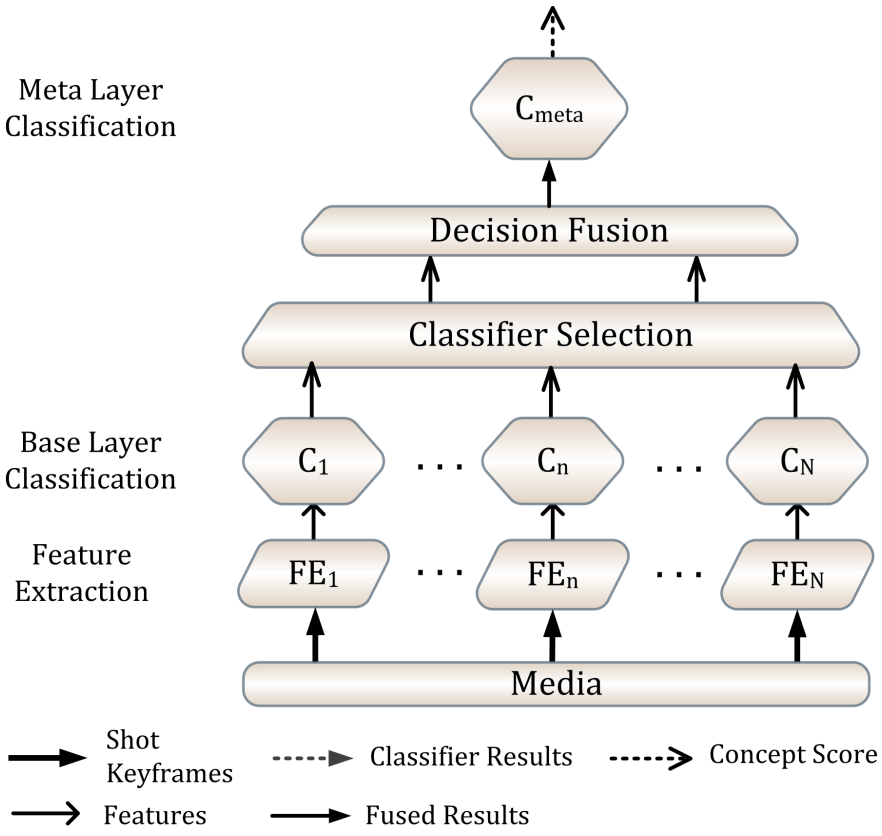

Figure 4.2: General structure of FSG-RO architecture.

fication, *Feature Selection*, *Decision Fusion* and *Meta Layer Classification*. In Feature Extraction module, features are extracted from video shots. In Base Layer Classification module the features are classified using fuzzy k-NN classifiers and membership values are produced for each video shot. In Feature Selection module, the classification results of the selected features are filtered and in Decision Fusion module membership values are combined using vector concatenation. Finally, in Meta Layer Classification module the fused membership values is classified using fuzzy k-NN classifier. The details of each module is explained in the following sections.

4.2.1 Feature Extraction

In this stage of the concept detection on a video shot sample vs_i, we first obtain a set of keyframes from vs_i. We employ a simple method that the frames of vs_i

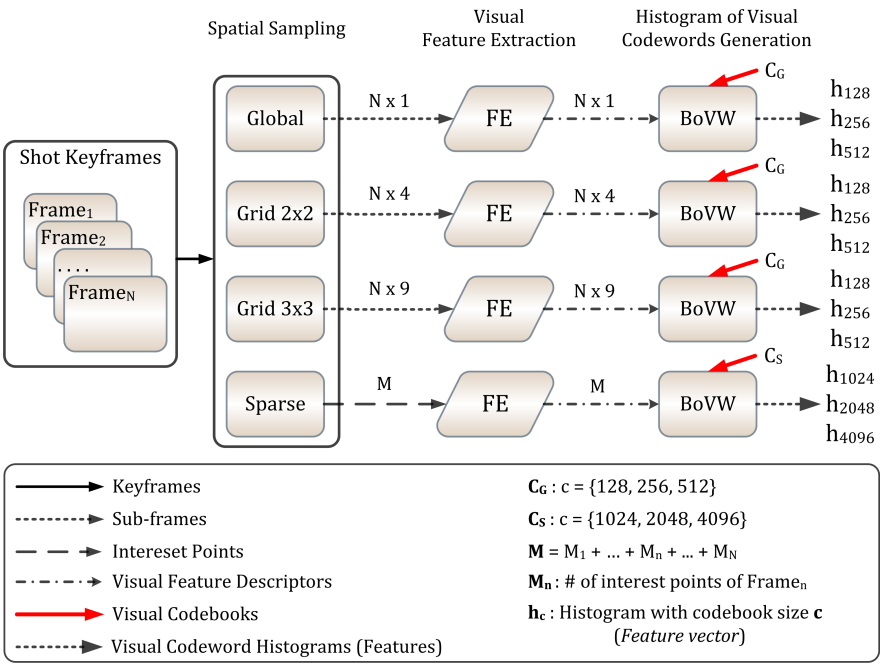

Figure 4.3: *Feature Extraction* process flowchart.

are uniformly sampled by using the equation:

$$\eta_i = \frac{T_i}{\delta_s} - 1, \tag{4.1}$$

where η_i is the number of frames obtained from vs_i, T_i is the duration of shot and δ_s is the time interval between two frames. Keyframes are taken at time instants $t_k = i.\delta_s$ for $k = 1, 2, \ldots, N_i$.

The set of keyframes obtained from vs_i are used as input for feature extraction process and the output is the set of histograms referred as *features*. The steps of this stage, namely *Spatial Sampling*, *Visual Feature Extraction* and *Visual Codewords Histogram Generation*, are shown in Figure 4.3 and explained in the following sections.

4.2.1.1 Spatial Sampling

In this step, different spatial sampling methods, namely, *Global* , $Grid2 \times 2$, $Grid3 \times 3$ and *Sparse*, are used in order to extract sub-frames and key points from each keyframe $F_j \in KF_i$, where $KF_i = \{F_j\}_{j=1}^{\eta}$ is the set of η keyframes extracted from a sample vs_i.

In the *Global* method, the whole frame is used and the output for vs_i is the set of frames $S_i^1 = \{f_j^1\}_{j=1}^{\eta}$ where $f_j^1 = F_j$ is the single sub-frame of $F_j, \forall j = 1, 2, \ldots, \eta$.

In the $Grid2 \times 2$ method, each keyframe F_j is segmented into 4 segments, each segment is a sub-frame, by using 2×2 cell. The output for vs_i is the set of sub-frames $S_i^2 = \{f_k^2\}_{k=1}^{4\eta}$ where S_i^2 contains 4 sub-frames for each keyframe F_j, $\forall j = 1, 2, \ldots, \eta$.

In the $Grid3 \times 3$ method, each keyframe F_j is segmented into 9 segments, each segment is a sub-frame, by using 3×3 cell. The output for vs_i is the set of sub-frames $S_i^3 = \{f_k^3\}_{k=1}^{9\eta}$ where S_i^3 contains 9 sub-frames for each keyframe F_j, $\forall j = 1, 2, \ldots, \eta$.

Finally, in the *Sparse* method, for video vs_i the set of interest points $S_i^P = \{p_k\}_{k=1}^{M}$ where S_i^P contains all the interest points extracted from each keyframe F_j, $M = M_1 + \ldots + M_j + \ldots + M_\eta$ and M_j is the number of interest points extracted from F_j. The number of interest point M_j is a frame-dependent parameter and may range from several hundreds to several thousands.

4.2.1.2 Visual Feature Extraction

Visual feature extraction process is achieved for vs_i by using the sub-frame sets, which are S_i^1, S_i^2, S_i^3, and the set of interest points which is S_i^P. The visual features $Features^G = \{$*Color Layout*, *Color Moments*, *Color Structure*, *Co-occurrence Texture*, *Dominant Color*, *Edge Histogram*, *Homogeneous Texture*, *Scalable Color*, *Wavelet Texture*$\}$ are extracted for each of the sub-frame sets and *SIFT* feature is extracted from the set of interest points.

For the sub-frame set S_i^1, for each visual feature $vf_j \in Features^G$, a set of descriptor vectors $D_{i,j}^1 = \{\overline{d_j}(f_k)\}_{k=1}^{|S_i^1|}$, where $\overline{d_j}(f_k)$ is the descriptor vector extracted from the subframe $f_k \in S_i^1$ for vf_j, is constructed.

For the sub-frame set S_i^2, for each visual feature $vf_j \in Features^G$, a set of descriptor vectors $D_{i,j}^2 = \{\overline{d_j}(f_k)\}_{k=1}^{|S_i^2|}$, where $\overline{d_j}(f_k)$ is the descriptor vector extracted from the subframe $f_k \in S_i^2$ for vf_j, is constructed.

For the sub-frame set S_i^3, for each visual feature $vf_j \in Features^G$, a set of descriptor vectors $D_{i,j}^3 = \{\overline{d_j}(f_k)\}_{k=1}^{|S_i^3|}$, where $\overline{d_j}(f_k)$ is the descriptor vector extracted from the subframe $f_k \in S_i^3$ for vf_j, is constructed.

Similarly, for the visual feature *SIFT*, the descriptor set $D_{i,s}^P = \{\overline{d_s}(p_k)\}_{k=1}^{|S_i^P|}$, where $\overline{d_s}(p_k)$ is the *SIFT* descriptor extracted from the interest point $p_k \in S_i^P$, is constructed.

As a result of this step, for a video shot vs_i we obtain 28 descriptor sets.

4.2.1.3 Histogram of Visual Codewords Generation

In this step, the visual feature descriptors are projected into another feature space, which is more robust and efficient for testing and training purposes, by utilizing the well-known *Bag-of-Visual-Words* (BoVW) [76] method, as explained below.

The visual descriptor set $D_{i,j}^R$, which contains the descriptor vectors of visual feature vf_j extracted from the sub-frame set $S_i^R \in \{S_i^1, S_i^2, S_i^3\}$ of vs_i, is projected to *visual codeword histogram* $\overline{h_{i,j}^R}(c)$ for the values of $c \in C_G = \{128, 256, 512\}$ which is the codebook size of the visual codebooks obtained in training phase using k-means clustering method. As a result, 3 different histograms are generated for a descriptor set.

Similarly, the keypoint descriptor set $D_{i,s}^P$ of vs_i is projected to visual codeword histogram $\overline{h_{i,s}^P}(c)$ for the values of $c \in C_S = \{1024, 2048, 4096\}$. For the descriptor set of *SIFT* feature, 3 different histograms are generated.

As a result of *Feature Extraction* process total of 84 different histograms are constructed for a single video shot vs_i.

In the rest of the thesis, these 84 histogram types will be referred as *feature types*. A *feature type* FE_j will be shown with 3-tuple list (vf, R, c) where $vf \in Features^G \cup \{SIFT\}$ is the visual feature, $c \in C_G \cup C_S$ is the codebook size and $R \in \{Global, Grid2\times2, Grid3\times3, Sparse\}$ is the spatial sampling method. The set of 84 feature types will be shown using $FE = \{FE_j\}_{j=1}^{84}$. For example, the feature $FE_1 = (ColorLayout, Global, 128)$ refers to the type of visual codeword histogram with codebook size 128, projected from the descriptors of visual feature *Color Layout* that are extracted from the sub-frames extracted using *Global* spatial sampling method. Also, the histogram generated for video shot vs_i by using feature type FE_j will be referred as $\overline{h}_{i,j}$.

4.2.2 Base Layer Classification

For each feature type $FE_j \in FE$ a fuzzy k-NN classifier C_j is employed as base layer classifier in FSG-RO. Each classifier C_j receives a set of visual codeword histograms of video shots $\{\overline{h}_{i,j}\}_{i=1}^{N}$, where $\overline{h}_{i,j}$ is extracted from a video shot vs_i obtained from a training set $VS = \{(vs_i, y_i)\}_{i=1}^{N}$ for each feature type $FE_j \in FE$. The output of a fuzzy k-NN classifier is a membership value $0 \leq \mu(\overline{h}_{i,j}) \leq 1$ measuring the degree of the membership that the searched concept is available in video shot vs_i. The membership value is computed by;

$$\mu(\overline{h}_{i,j}) = \frac{\sum_{n=1}^{k} y_{l(n)} (\left\| \overline{h}_{i,j} - \overline{h}_{l(n),j} \right\|)^{-\frac{2}{\varphi-1}}}{\sum_{n=1}^{k} (\left\| \overline{h}_{i,j} - \overline{h}_{l(n),j} \right\|)^{-\frac{2}{\varphi-1}}} \tag{4.2}$$

where $l(n)$ is the index of n^{th} nearest neighbour $\overline{h}_{l(n),j}$ of $\overline{h}_{i,j}$ and $y_{l(n)} \in \{0, 1\}$ is the label of $vs_{l(n)}$. The label $y_{l(n)}$ is 1 for the histograms of positively labelled video shots and 0 for others. The value φ is the fuzzification parameter, $\forall i = 1, 2, \ldots, N, \forall j = 1, 2, \ldots, J$ (in our system $J = 84$).

In the training phase of the proposed FSG-RO method, the membership value $\mu(\overline{h}_{i,j})$ of each shot vs_i is computed by using leave-one-out cross validation tech-

nique for each $(\overline{h}_{i,j}, y_i)$ in the validation set $VS_j^{cv} = VS_j - (\overline{h}_{i,j}, y_i)$, where $VS_j = \{(\overline{h}_{i,j}, y_i)_{i=1}^{N}\}$.

In the test phase, the membership value $\mu(\overline{h'}_{i,j})$ of each test shot vs'_i obtained from the test set $VS' = \{vs'_i\}_{i=1}^{N'}$ is computed by using the equation (4.2) with a set of test codeword histograms $VS'_j = \{\overline{h'}_{i,j}\}_{i=1}^{N'}$ and train histograms VS_j in each classifiers $C_j, \forall j = 1, 2, \ldots, J$.

4.2.3 Classifier Selection

In this stage, classifiers to use in meta layer classification are filtered. The details of the selection of classifiers will be explained in section 4.3. In this part, the classifier selection process in training and test phases will be briefly explained.

A set of fuzzy k-NN classifiers $C^s = \{C_{f(d)}\}_{d=1}^{D}$, where D is the number of selected classifiers, $f(d) \in \{1, 2, \ldots, J\}$ is the id of d^{th} selected classifier $C_{f(d)}$ and $C^s \subseteq \{C_j\}_{j=1}^{J}$, is determined by the feature selection methods using a set of validation shots(see section 4.3).

In training phase, the membership values $\mu(\overline{h}_{i,j})$ computed by classifier C_j for each train shot vs_i obtained from $VS = \{vs_i\}_{i=1}^{N}$ and is passed to the *Decision Fusion* stage if and only if the feature type $C_j \in C^s, \forall j = 1, 2, \ldots, J$.

In testing phase, the membership value $\mu(\overline{h'}_{i,j})$ computed by classifier C_j for each test shot vs'_i from $VS' = \{vs'_i\}_{i=1}^{N'}$ is passed to *Decision Fusion* stage if and only if the classifier $C_j \in C^s, \forall j = 1, 2, \ldots, J$.

4.2.4 Decision Fusion

In decision fusion stage, selected classifiers are fused by using vector concatenation operation. The training and test phases are explained below:

In the training phase, for each shot vs_i a single vector $\overline{\mu}(vs_i) = [\mu(\overline{h}_{i,f(d)})]_{d=1}^{D}$ is constructed, where D is the number of classifiers selected in *Classifier Selection* stage, $f(d) \in \{1, 2, \ldots, J\}$ is the id of d^{th} selected classifier $C_{f(d)} \in C^s$, $\mu(\overline{h}_{i,f(d)})$

is the membership value for histogram $\overline{h}_{i,f(d)}$ of vs_i. $\mu(VS) = \{\overline{\mu}(vs_i)\}_{i=1}^{N}$ is passed to meta classification layer as training data.

In testing phase, similar to training phase, for each shot $vs'_i \in VS'$ a membership vector $\overline{\mu}(vs'_i) = [\mu(\overline{h}_{i,f(d)})]_{d=1}^{D}$ is generated and $\mu(VS') = \{\overline{\mu}(vs'_i)\}_{i=1}^{N'}$ is passed to meta classification layer as testing data.

4.2.5 Meta Layer Classification

Fuzzy k-NN classifier C_{meta} is employed as a meta layer classifier for fusion of the decisions of base layer classifiers. The set of membership vectors $\mu(VS)$ obtained in decision fusion stage for training shots is utilized as training data in C_{meta}. The set of membership vectors $\mu(VS')$ generated for test shots is classified by C_{meta} using the equation 4.2. For each test shot vs'_i obtained from $VS' = \{(vs'_i, y_i)\}_{i=1}^{N'}$ a final membership value $\mu_f(vs'_i)$, measures the probability that video shot vs'_i contains the concept, is computed.

4.3 Classifier Selection

In order to determine the feature types to be used in meta layer classification, a new set of video shots $VS = \{(vs_i, y_i)\}_{i=1}^{N}$, named as validation set, is used. For each vs_i obtained from VS a membership value $\mu(\overline{h}_{i,j})$ is computed by using (4.2) with $VS_j = \{\overline{h}_{i,j}\}_{i=1}^{N}$ as test data and VS_j as training data in each classifier C_j.

For each classifier C_j, we produce a rank list:

$$Rank_j := [(vs_{l(1)}, y_{l(1)}), \ldots, (vs_{l(i)}, y_{l(i)}), \ldots, (vs_{l(N)}, y_{l(N)})] \tag{4.3}$$

where $l(i)$ is the index of the shot $vs_{l(i)}$ with i^{th} highest membership value in $\mu_j = \{\mu(\overline{h}_{i,j})\}_{i=1}^{N}$ which is the membership values produced by C_j for each shot of VS, is generated by using validation shots, $\forall i = 1, 2, \ldots, N$, $\forall j = 1, 2, \ldots, J$.

The rank list $Rank_j$ of each base layer classifier C_j are given as input to the

classifier selection method explained in following section to determine a set of classifiers $C^s = \{C_{f(d)}\}_{d=1}^{D}$, where D is the number of selected classifiers, $f(d) \in \{1, 2, \ldots, J\}$ is the id of d^{th} selected classifier $C_{f(d)}$ and $C^s \subseteq \{C_j\}_{j=1}^{J}$,

In the following sections, firstly the motivation of selecting the features will be explained. Later the details of the feature selection method will be explained.

4.3.1 Motivation: Why to Select Classifiers?

Fusing all the outputs of base-layer classifiers is not a feasible approach for many reasons. First of all, some features and the classifiers related with these features may be statistically dependent and redundant, causing unnecessary computational cost. Secondly, feature types may have variety of different representation power for each class. In other words some features may be very effective to represent some specific classes, while may fall short to represent some other classes. For example, some color features have a strong power to represent *Air* classes. However, these features are not suitable to represent *Crowds*. Therefore, one needs to select the features which complement each other such that they represent all the classes collectively.

In order to boost the performance of a concept detection system, classifiers related with the subset of complementary features are to be selected. Combining large number of classifiers from various domains in the FSG may be resulted with performance loss. For these cases, using the classifiers of complementary features rather than all classifiers is required to boost the performance of the base-layer classifiers. This is the reason why we utilize classifier selection method, rather than combining all classifiers in our FSG-RO system. Additionally, selection of classifiers reduce the time and space complexity of training and testing phases of a system.

4.3.2 Classifier Selection Algorithm

The set of rank lists $Rank = \{Rank_j\}_{j=1}^{J}$, where $Rank_j$ is the rank list generated by using classifier C_j on validation data VS, is taken as input. The *Mean*

Average Precision (mAP) [48] performance of each classifier C_j is calculated by using $Rank_j$:

$$mAP_j = (\frac{1}{N_p}) \sum_{i=1}^{N_p} \frac{i}{o_{i,j}} \tag{4.4}$$

where N_p is the number of shots labelled as positive ($y_i = 1$) in $Rank_j$ and $o_{i,j}$ is the order of the i^{th} positively labelled sample in the $Rank_j$.

Algorithm 1: Classifier selection algorithm of FSG-RO.

input : N the number of classifiers to select and rank list $Rank_j$ of classifier C_j, $\forall j = 1, 2, \ldots, J$

output: Set of selected classifiers C^s

```
foreach j = 1, 2, ..., J do
    Calculate mAP_j;
end
C^s = {C_{m(i)}}_{i=1}^{N}, the classifier C_{m(i)} has the i^{th} highest mAP;
```

The classifier selection algorithm of FSG-RO is given in 4. Simply, our algorithm computes the mAP performance of each individual base layer classifier and selects the first N classifiers with the highest mAP performance.

As mentioned in *Classifier Fusion* section, there should be also an independency between classifiers and using the best classifiers may not be enough to boost the performance of individual classifiers. In FSG-RO, individual base layer classifiers have some form of independency, since they trained for a wide variety of features which represent different characteristics of samples such as color and texture. Additionally, these features are extracted using different codebook sizes and spatial sampling methods that brings independency to classifiers. As a result, independency of combined classifiers is also achieved in FSG-RO and combining best classifiers can be used to boost the performance of individual classifiers.

CHAPTER 5

EXPERIMENTAL RESULTS

The performance of the FSG-RO architecture is tested with experiments that are carried out on broadcast media domain. The visual concepts *Air*, *Artificial Edge*, *Crowd*, *Fire*, *Grass*, *Soil*, *Water Image* are classified using FSG-RO architecture and the ranking performances of the FSG-RO for the given concepts are presented comparatively with the performances of individual base layer classifiers and state-of-art ensemble learning techniques.

In the following sections, firstly, the information about the datasets used for training, validation and testing purposes will be explained. Then, the tests for the training step together with the experiments performed on the classifier selection process will be analysed. Finally, results of the tests will be discussed.

5.1 Description of the Dataset

The datasets used in the experiments contain video shot samples that are mostly recorded from the broadcasts of the Turkish national television (TV) network. Each video shot in these datasets is labelled by human operators according to normative semantic definitions. For each concept, the number of positively and negatively labelled video shots in the train, validation and test datasets are shown in Table 5.1.

The training sets, employed by the FSG-RO are the subsets of the training sets employed by the KavTan system [56] while test and validation sets of FSG-RO are the subsets of the test sets employed by the KavTan system. Note that

Table 5.1: Number of positive and negative video shots in datasets

	Train Set		Validation Set		Test Set	
Concept	Positive	Negative	Positive	Negative	Positive	Negative
Air	589	1776	353	2022	354	2023
Artificial Edge	243	729	868	2351	869	2351
Crowd	2433	7299	170	1836	171	1836
Fire	999	2997	206	1493	207	1493
Grass	332	996	279	2131	279	2131
Soil	274	820	191	1525	191	1525
Water Image	616	1848	163	1762	164	1763

the number of samples for each class in the train, test and validation datasets varies in a wide range. This is basically because of the fact that the training sets are extracted in KavTan project had imbalanced distribution of the concepts. However, the tests sets are formed for this study is rather balanced. Since our aim in this study is to boost the performance of the base layer classifiers, we do not pay attention to the imbalance train set problem. In our experiments, for the training sets the ratio $\frac{\#\ of\ positive\ shots}{\#\ of\ negative\ shots}$ is taken as $\frac{1}{3}$ for the effectiveness of fuzzy k-NN classifiers. Additionally, we divide the test sets of the KavTan into two sets, namely validation and test sets. Validation set is used to estimate the k-parameter of the k-NN algorithm. We, also select features using the validation set.

5.2 Training of the Proposed FSG-RO

Since FSG-RO is designed as a binary classification system, for each concept, a separate FSG-RO system is trained as a dichotomizer. In the training of base layer classifiers of a FSG-RO, the only the parameter k for fuzzy k-NN classifiers is defined. In order to pay more attention to the classifier selection method and meta layer classifier, we select the $k = 5$ for all base-layer classifiers. For meta layer classifier, additional to the k parameter, we also select a *best set of classifiers*. Brief explanation of classifier selection and estimation of k for meta layer classifier is explained in following section.

5.2.1 Classifier Selection and Estimation of k for Meta-Layer Classifier

For each concept $Co \in Concepts$, where $Concepts = \{Air, Artificial\ Edge, Crowd, Fire, Grass, Soil, Water\ Image\}$, a seperate FSG-RO system is trained. For each system, the parameter pair (n, k) where n is the number of classifiers to be selected by classifier selection method and k is the count of nearest neighbours utilized by the meta fuzzy k-NN classifier, is determined experimentally. For each FSG-RO system trained for a concept $Co \in Concepts$, we tested each parameter pair $(n, k) \in Parameters^{Co}$;

$$Parameters^{Co} = N \times K^{Co}, \tag{5.1}$$

where $N = \{2, 3, \ldots, \frac{J}{2}\}$, J is the number classifiers in base layer and $K^{Co} = \{k_i^{Co}\}_{i=1}^{50}$ is the set of k values that $k_i^{Co} = \frac{i}{100} \times |S_{Co}^{tr}|$ and $|S_{Co}^{tr}|$ is the number of train samples of concept Co.

Briefly, in experiments, at most half of the classifiers are selected as the input to the meta layer classifier. For the parameter k, since the number of train samples of each concept differs, we selected the values by considering the train set sizes of concepts. The set K^{Co} contains the k values from 1% to 50% of train set size of the concept Co. After the experiments, the pair (n, k) that gives the best performance is selected to train FSG-RO for the testing stage.

In the experiments, the performance of the system trained with parameter pair (n, k) is computed using the the mean average precision (mAP) 4.4 metric and shown with $mAP(n, k)$. The effects of parameters n and k on the performance are analysed separately.

In Figure 5.1, the performance of the system is shown for different classifier counts n. The performance of the system for classifier count $n \in \{2, 3, \ldots, 42\}$ is computed using

$$mAP^n = \frac{\sum_{j=1}^{50} mAP(n, k_j)}{50}, \tag{5.2}$$

that is the average performance of the systems trained by selecting n features.

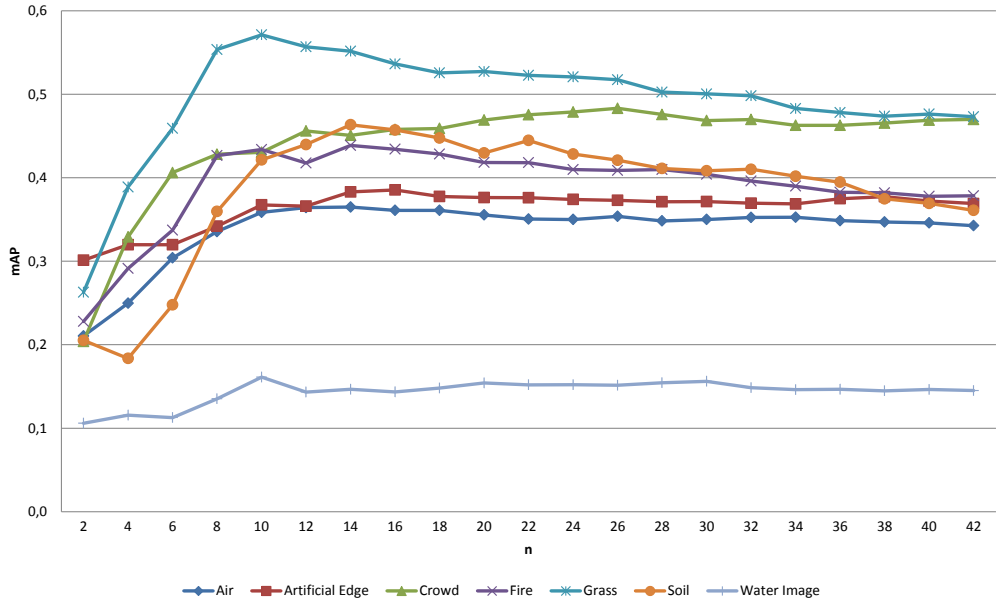

Figure 5.1: Average performance of FSG-RO systems which fuse n classifiers.

For most of the concepts best performances are obtained for the small values of n which are between [10, 15]. Since the dimensions used for the experiments are relatively small, and the concept *Crowd* has good performance for greater values of n, we think that there is no curse of dimensionality problem. The reason of performance decrease for the $n > 10$ is that the classifiers which have high performances are firstly selected by classifier selection and the performance increases up to a point. Then, as n increases our method selects the classifiers that have poor performance when compared to selected ones. This fact results in a decrease. Therefore, combining the classifiers with high performance also increases the performance of the system.

In Figure 5.2, the performance of the system is shown for different values of k of fuzzy k-NN method. The performance of the system for the value $k = k_i$ is calculated using $mAP^{k_i} = \frac{\sum_{n=2}^{J/2} mAP(n,k_i)}{J/2-1}$ that is the average performance of the systems trained using parameters (n, k) where $k = k_i$ and $n = 2, 3, \ldots, fracJ2$. Although, the best performances, generally, obtained for the values of k between 5% and 15% of the train set size, the performance changes are very little for greater values of k. This is the expected situation for fuzzy k-NN algorithm,

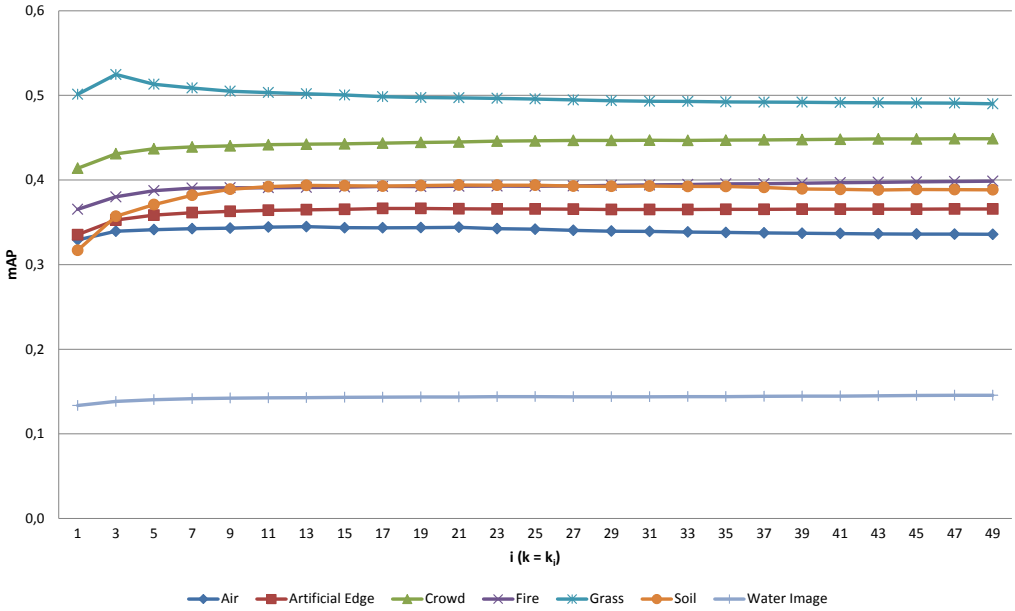

Figure 5.2: Average performance of FSG-RO systems which trained using $k = k_i$.

since the nearest samples have more weight for determining the probabilistic decision score and the ones far away have less weights. On the other hand, we observe a slight increase in the performances of the concepts *Crowd* and *Fire*. The common point of these concepts is that their train set sizes are greater than the other train sets. It is observed that the classes with high number of samples used in fuzzy k-NN, results in slight changes on the decision of nearest samples.

5.3 Experiments

In experiments, for each concept the performances of base layer classifiers and the overall FSG-RO system are computed. Additionally, in order to better analyse the results of FSG-RO system, we define a new method, called $Combine_{ALL}$, and compute its performance for each concept. In the $Combine_{ALL}$ method, simply we cancel the classifier selection stage of FSG-RO and utilize the classifier results in meta layer for the detection.

Moreover, FSG-RO system performance is compared with SVM which classifies the aggregated feature space $FE = FE_1 \times FE_2 \times \ldots FE_j \ldots FE_J$. For each

concept, we implement SVM with the RBF kernel by using LIBSVM [11]. For learning each SVM classifier, we determine γ parameter for RBF kernel and C parameter for SVM model [11] by testing parameters, which are used in the study of Chang et al. [20], $C = \{2^0, 2^2, 2^4, 2^6, 2^8\}$ and $\gamma = \{2^{-4}g, 2^{-2}g, g, 2^2g, 2^4g\}$ where $g = \frac{1}{D_f}$ and D_f is the dimensionality of the aggregated feature space. The SVM model which gives the best mAP performance on validation set is selected for testing purposes.

Finally, we compare FSG-RO with RReliefF [41] feature selection method. By using RReliefF, the most important L attributes of the aggregated feature space are selected and a SVM with RBF kernel is trained for new feature space. The parameters $L = \{2^5, 2^6, 2^7, 2^8, 2^9, 2^{10}, 2^{11}, 2^{12}\}$, $C = \{2^0, 2^2, 2^4, 2^6, 2^8\}$ and $\gamma = \frac{1}{D_f}$ are tested for training RReliefF system. Best parameters which give the best mAP performance on validation set is selected for testing purposes.

The performance for each concept are analysed in subsequent sections and in following items some notes about the results are given.

- For each concept, the mAP of individual classifiers, $Combine_{ALL}$, FSG-RO, RReliefF and SVM methods are shown in Figures 5.3 - 5.9.
- We compute the ratio of positive samples in the test set of each concept, $Base = \frac{\#\ of\ positive\ test\ samples}{\#\ of\ test\ samples}$ and it is shown in Figures 5.3 - 5.9. We use the value of $Base$ to see the success of classifiers. This line can be taken as the performance of random ranking.
- In Figures 5.3 - 5.9, each base layer classifier C_j trained for feature $FE_j = (vf, R, k)$ where vf is the visual feature, R is the spatial sampling method and k is the codebook size used for histogram generation is shown by using the abbreviations of its visual feature and spatial sampling method. The abbreviatons for visual features are *CL (Color Layout)*, *CM (Color Moments)*, *CS (Color Structure)*, *CoT (Cooccurence Texture)*,*DC (Dominant Color)*, *EH (Edge Histogram)*, *HT (Homogeneous Texture)*, *SC (Scalable Color)*, *WT (Wavelet Texture)* and for spatial sampling methods are 1×1 ($Global$), 2×2 ($Grid2 \times 2$) and 3×3 ($Grid3 \times 3$).

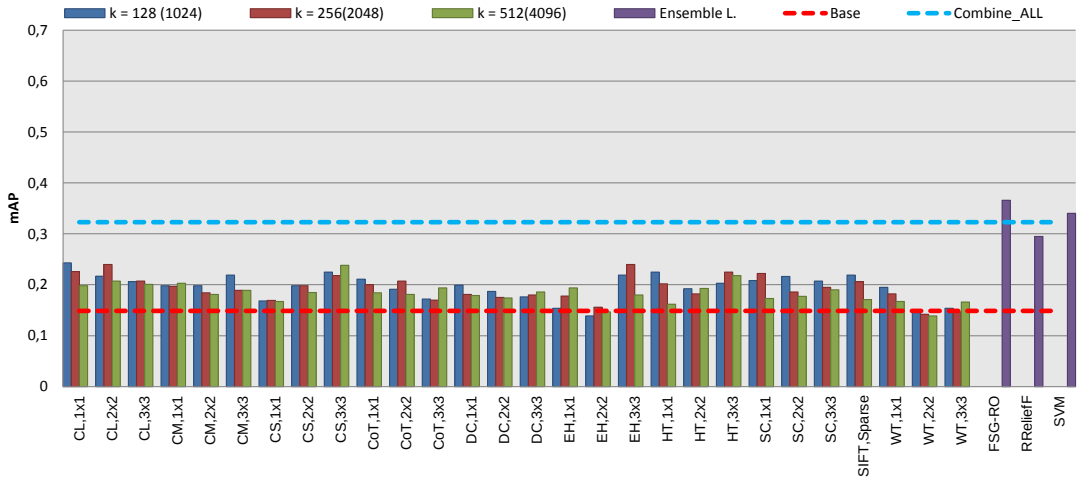

Figure 5.3: *Air* concept detection performances.

5.3.1 Air

The performances of base layer classifiers for *Air* concept are generally greater than the *Base* performance. However, there are no prominent performances between them. The color and edge characteristics are discriminative features for *Air*. Additionally, *Air* concept is mostly located in the upper regions of the samples. As a result, using regional features also increases the performance of the system. As it can be seen in Figure 5.3, generally best individual performances are obtained for color features extracted by using $Grid2 \times 2$ and $Grid3 \times 3$ sampling methods. Also, $EdgeHistogram$ for $Grid3 \times 3$ sampling method gives better performances than other classifiers.

As shown in Figure 5.3, FSG-RO system improves the best individual classifier performance and outperforms the $Combine_{ALL}$ method. This shows that FSG-RO architecture successfully achieves the classifier selection and fusion for *Air* concept. Since there are various type of base layer classifiers which achieve better performances that *Base* performance, FSG-RO can be able to improve the performance by combining best classifiers. Moreover, FSG-RO outperforms both RReliefF and SVM methods which also gives better performances than individual classifiers.

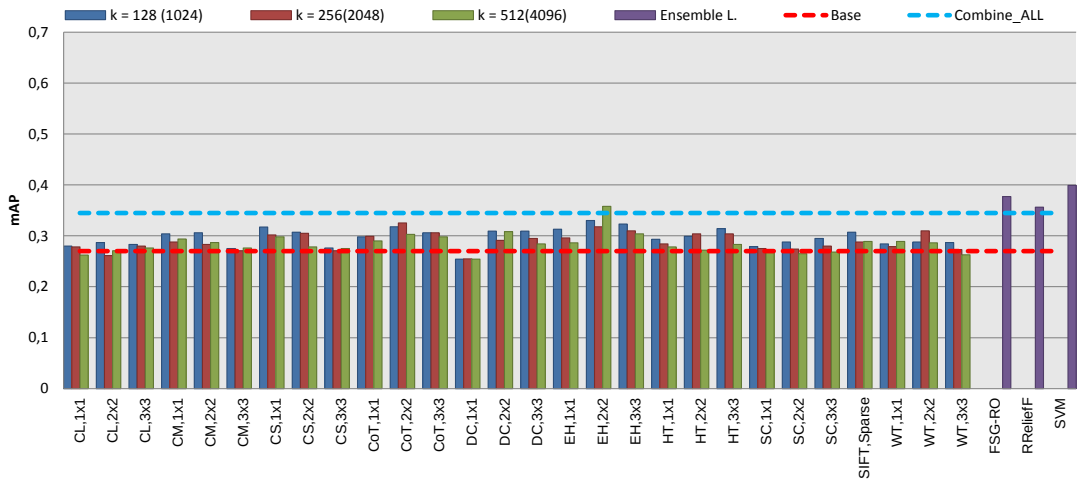

Figure 5.4: *Artificial Edge* concept detection performances.

5.3.2 Artificial Edge

The performances for *Artificial Edge* are shown in Figure 5.4. The base layer classifier performances are generally slightly different than *Base* performance. *Artificial Edge* concept may have wide variety of colors and texture characteristics in video shots, as a result, the classifiers for color and texture features can not achieve significant performances. Only the $EdgeHistogram$ feature achieves considerably better performance than *Base* performance. This is an expected situation since the edge distribution of nature are mostly different than man-made items which has smooth artificial edges.

As shown in Figure 5.4, FSG-RO system slightly boost the performances of best individual classifier $Combine_{ALL}$ method. Since, most of the individual classifiers are weak and can not significantly boost the random performance, FSG-RO can not improve the performance as for the *Air* concept. FSG-RO, also boost the RReliefF method while SVM method achieves the best performance. On the other hand, the performance of SVM is also slightly better than FSG-RO and than that of the best individual classifier.

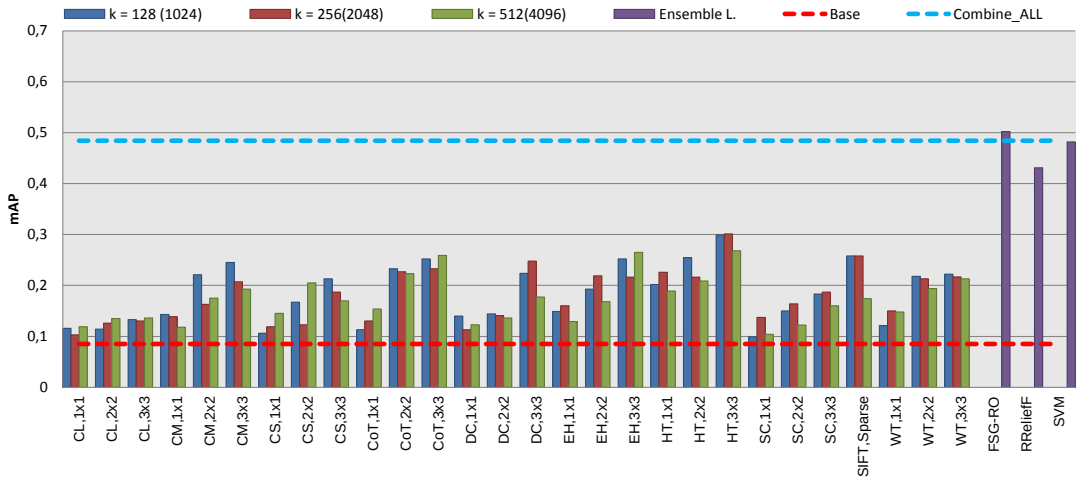

Figure 5.5: *Crowd* concept detection performances.

5.3.3 Crowd

The performances for *Crowd* are shown in Figure 5.5. All of the base layer classifiers achieve better performances than *Base* performance. However no prominent performances are achieved by individual classifiers. The view variations, such as, variation in clothes of people and varying density of people make a challenging task to detect crowd. Generally, texture and SIFT features achieve better performances than others. Because of the nature of crowd, it produce high number of interest points and its texture is different from nature and artificial textures.

For *Crowd* concept, prominent results are obtained by combination methods. The least improvement achieved by RReliefF is 13% while FSG-RO achieves 20% improvement on the performance. Between combination methods, FSG-RO achieved the best performance. It slightly boost the $Combine_{ALL}$ and SVM methods which achieve nearly same performance. This shows the effectiveness of classifier selection for *Crowd* concept.

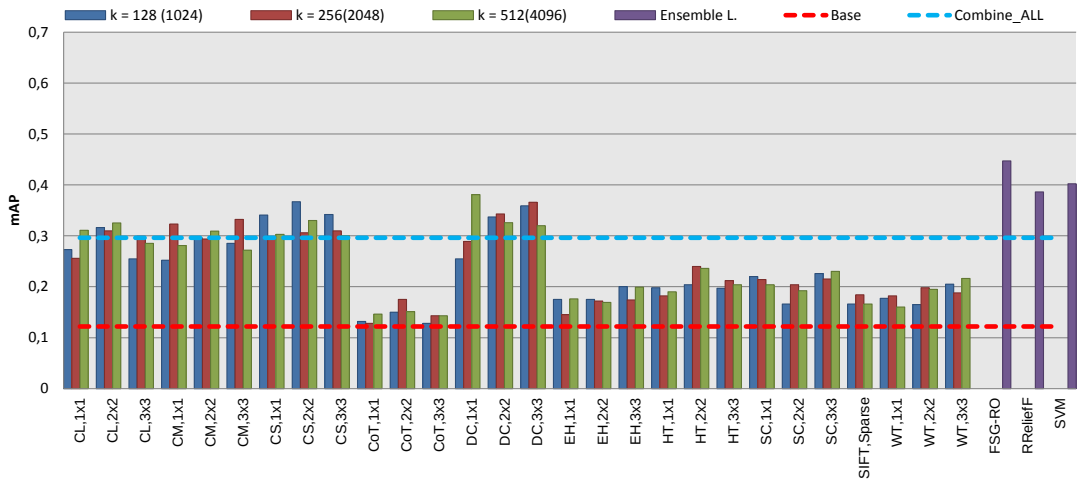

Figure 5.6: *Fire* concept detection performances.

5.3.4 Fire

The performances of base layer classifiers for *Fire* concept are shown in Figure 5.6. Since, the red and yellow are main colors of fire, individual classifiers of color features achieve prominent results. Also, texture of fire is another source to discriminate fire from other concepts and classifiers of textures feature obtain performances than *Base* performance. Generally, *Fire* concept prensent locally in video frames that only a part of a frame contains it. As a result, employing local features by using spatial sampling methods improves the performance of the classifiers. As it can be seen in figure, the classifiers of a visual feature trained for $Grid2 \times 2$ and/or $Grid3 \times 3$ sampling methods have better performances than the classifiers of same visual feature which are trained for *Global* (1×1) sampling method.

For *Fire* concept FSG-RO system outperforms the performances of the best individual classifier, $Combine_{ALL}$, RReliefF and SVM methods.

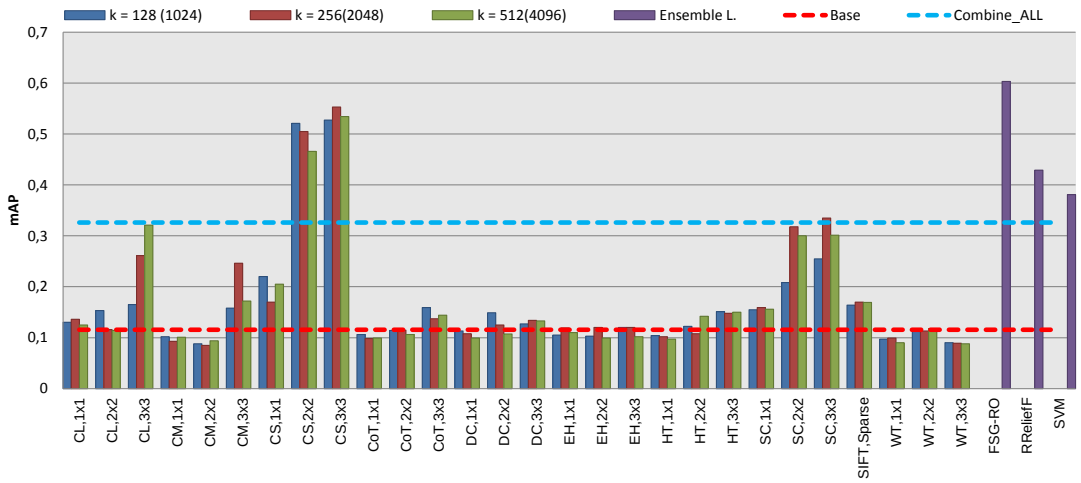

Figure 5.7: *Grass* concept detection performances.

5.3.5 Grass

The green color of the *Grass* concept is the main characteristic that discriminate *Grass* from other concepts. Additionally, grass is mostly located in the lower regions of the samples. As a result, the individual base layer classifiers of local color features achieve better performances than other classifiers as shown in figure 5.7.

For *Grass* concept, FSG-RO is the only method that boost the performance of the best individual base layer classifier. $Combine_{ALL}$, RReliefF and SVM methods can not improve the system performance, rather, they decrease the performance. In $Combine_{ALL}$ method, combining all classifiers decrease the system performance. This shows the importance of classifier selection stage of FSG-RO which select the best classifiers to optimize system performance.

5.3.6 Soil

The performances for *Soil* are shown in Figure 5.8. Generally, soil located in the lower regions of video frames. As a result, best performances are achieved by local feature classifiers and key point based classifiers. On the other hand,

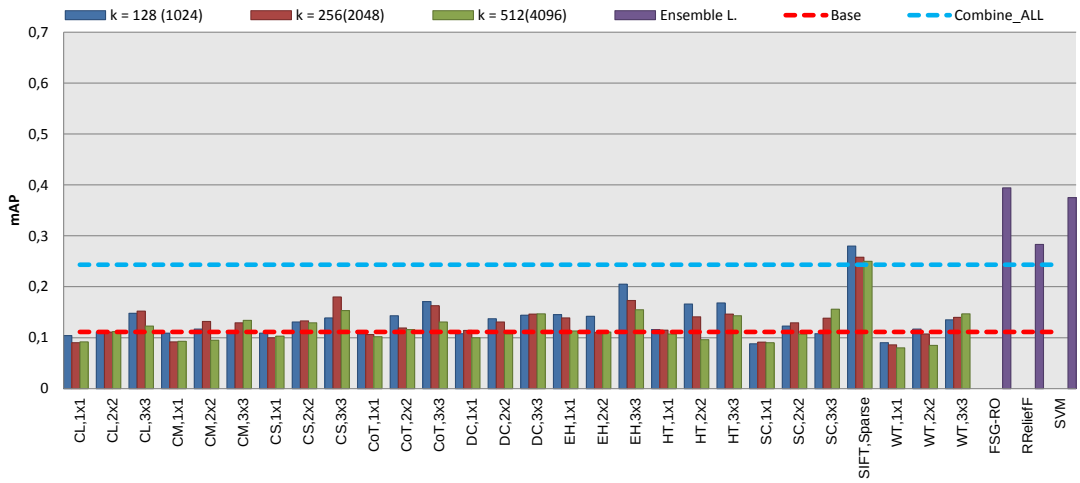

Figure 5.8: *Soil* concept detection performances.

Soil concept mostly present together with the items like grass, tree and water in videos. In such an enviroment, soil seems as a small part of a frame and can not be effectively determined. Because of these reasons, other than the classifiers trained for $Grid3\times3$ sampling and $SIFT$ feature, most of the individual classifier performances can not boost the *Base* performance as shown in Figure 5.8.

Despite of the poor performances of the base layer classifiers, FSG-RO significantly improves the performance of the system by combining best classifiers. Also, FSG-RO boost the $Combine_{ALL}$, RReliefF and SVM methods.

5.3.7 Water Image

The performances for *Water Image* are shown in Figure 5.9. Most of the individual classifiers have lower performances than *Base* performance. The appearance of water depends to its enviroment. In a nature scene trees and mountains will be appear on water, on the other hand, in a city scene buildings will be appear. As a result, color and texture features are insufficent to detect water images.

Since, the base-layer classifier performances are lower than *Base* performance, FSG-RO system can not outperforms the best individual classifier. On the other hand, FSG-RO achieves better performance than $Combine_{ALL}$ method.

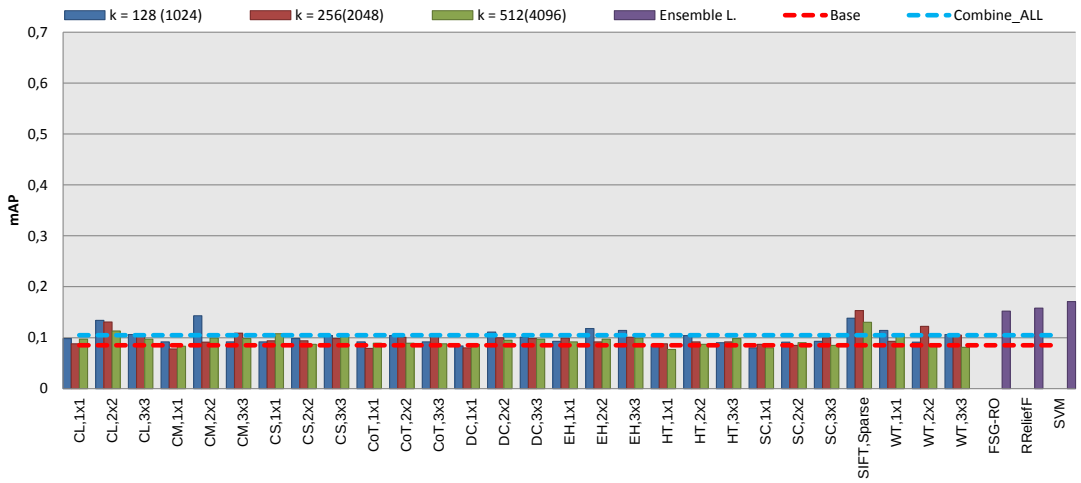

Figure 5.9: *Water Image* concept detection performances.

This shows the effectiveness of classifier selection. SVM and RReliefF methods slightly boost the system performance. Similar to *Artificial Edge* concept, when the base layer classifier performances are poor, feature fusion approach of SVM and RRelieF methods generally achieves better performances.

5.3.8 Chapter Summary

For each concept, the performances of best individual classifier, $Combine_{ALL}$, FSG-RO, RReliefF and SVM method are shown in Table 5.2. For all concept, FSG-RO systems successfully boost the performance of $Combine_{ALL}$ method. Especially, for the concepts *Fire, Grass, Soil, Water Image*, employing classifier selection method gives much more successful results than combining all classifiers. This shows that classifier selection stage of FSG-RO has a significant contribution to solve the black art problem.

For the concepts *Air*, *Crowd*, *Fire*, *Grass* and *Soil*, FSG-RO system significantly boost the performance of the best individual classifiers. The common property of these concepts is that, there are several classifiers that shows higher performance than *Base* performance and these are from various spatial categories and various feature types.

Table 5.2: Performances of the best base layer classifier C_{best}, $Combine_{ALL}$, $FSG-RO$, $RReliefF$ and SVM methods.

Concept	C_{best}	$Combine_{ALL}$	FSG-RO	RReliefF	SVM
Air	0.243	0.323	**0.366**	0.295	0.340
Artificial Edge	0.358	0.345	0.377	0.356	**0.399**
Crowd	0.301	0.484	**0.502**	0.431	0.482
Fire	0.381	0.296	**0.447**	0.386	0.402
Grass	0.553	0.326	**0.603**	0.429	0.381
Soil	0.28	0.243	**0.394**	0.283	0.375
Water Image	0.153	0.105	0.152	0.158	**0.171**

Water Image is the only concept that FSG-RO can not boost the base layer classifier performances. As shown in Figure 5.9, similar to *Artificial Edge* concept, most of the base layer classifiers have very poor performances. As a result, boosting the best individual classifier by combining classifiers can't be achieved by FSG-RO. On the other hand, FSG-RO approximately shows the same performance with the best base layer classifiers.

Moreover, for the concepts *Air*, *Crowd*, *Fire*, *Grass* and *Soil*, FSG-RO outperforms the RReliefF and SVM methods. Especially, for the *Grass* concept, FSG-RO gives prominent results and when compared to RReliefF and SVM methods, respectively, it achieves 17% and 22% better performances. On the other hand, for the *Artificial Edge* and *Water Image* concepts SVM method slightly boost the FSG-RO method. It is obsaerved that, when the base layer classifier performances are poor, SVM method achieves a better performance.

CHAPTER 6

CONLUSION

In this thesis, a Stacked Generalization (SG) architecture is proposed for visual concept detection.

In the proposed architecture, called as Fuzzy Stacked Generalization Ranking Optimizer (FSG-RO), fuzzy k-NN classifiers are employed in base-layer. Then, a feature selection algorithm is used for selecting the feature types to combine in meta layer. For this purpose, the ranking performances of individual base-layer classifiers are computed on a validation data by using mean average precision (mAP) metric and feature types are selected according to their classification performances. Finally, the decisions for the selected feature types are fused by vector concatenation and the fused vector is classified by a fuzzy k-NN meta classifier.

FSG-RO architecture is tested on broadcast media data for the detection of visual concepts *Air, Artifical Edge, Crowd, Fire, Grass, Soil* and *Water Image*. The results are compared with the state of the art techniques which are SVM and RReliefF. Additionally, the performance of FSG-RO is compared with the performances of best individual base-layer classifier and $Combine_{ALL}$ method which is the case that FSG-RO is used without classifier selection stage and all decisions are combined in meta-layer.

In the experiments, for the concepts *Air, Artificial Edge, Crowd, Fire, Grass* and *Soil* FSG-RO boosts the performances of best individual base-layer classifiers and outperforms the $Combine_{ALL}$ method. Only, for the concept *Water Image*

FSG-RO outperforms $Combine_{ALL}$ method but achieves approximately same performance with the best individual base-layer classifier.

It is observed that when the individual base-layer classifiers are strong, FSG-RO achieves improvements on the performance of base layer classifiers. Especially, for the concepts *Air*, *Crowd* and *Soil*, respectively, 12%, 22% and 12% performance improvements are achieved with respect to best individual classifier performances.

Moreover, outperforming $Combine_{ALL}$ method for all concepts shows the effectiveness of the classifier selection method of FSG-RO. Especially, when compared to the $Combine_{ALL}$ method, FSG-RO achieves 28%, 15% and 16% performance optimization for the concepts *Grass*, *Fire* and *Soil*, respectively.

When compared to the state of the art techniques, FSG-RO also achieves comparable performances. For the concepts *Air, Crowd, Fire, Grass* and *Soil* FSG-RO outperforms the SVM and RReliefF methods. For the concept *Grass*, respectively, 22% and 17% better performances are obtained when compared to SVM and RReliefF methods. Also, for *Soil* concept FSG-RO achieves 11% better performance than RReliefF. On the other hand, for the concepts *Artificial Edge* and *Water Image* SVM achieves slightly better performances and for *Water Image* RReliefF slightly outperforms FSG-RO. The results show that FSG-RO is comparable with state of the art techniques.

The advantage of FSG-RO over the other methods is that for each concept FSG-RO selects classifiers to optimize the ranking performance. Also, each base layer classifier can be trained on different feature space. Therefore, different information obtained from different features and modalities can be used efficiently in FSG-RO. However, there is a challenge of FSG-RO that when the base layer classifiers have poor performances, other state of the art methods achieve best performances and FSG-RO can not boost the performance of best individual base layer classifier.

As future work, the classifier selection method of FSG-RO can be improved to achieve better performances for the cases that the base layer classifiers have

poor performances. Another future work is that, FSG-RO can be implemented as a multi-class classification system so that only one train step will be enough for the whole system and modelling relations between concepts can be used to improve the performance of the system. As a final future work, we can test FSG-RO on various datasets in the literature.

REFERENCES

[1] Z. Ahmad and J. Zhang. Selective combination of multiple neural networks for improving model prediction in nonlinear systems modelling through forward selection and backward elimination. *Neurocomputing*, 72(4-6):1198–1204, January 2009.

[2] E. Akbas and F. Yarman Vural. Automatic image annotation by ensemble of visual descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, June 2007.

[3] A. Amir, M. Berg, S.-f. Chang, G. Iyengar, C.-y. Lin, A. P. Natsev, C. Neti, H. Nock, M. Naphade, W. Hsu, J. R. Smith, B. Tseng, Y. Wu, D. Zhang, and I. T. J. Watson. IBM research TRECVID-2003 video retrieval system. In *In NIST TRECVID-2003*, 2003.

[4] A. F. D. Araujo, F. Silveira, H. Lakshman, J. Zepeda, A. Sheth, and B. Girod. The Stanford / Technicolor / Fraunhofer HHI Video. In *Proceedings of the TRECVID Workshop*, 2012.

[5] P. Atrey, M. Maddage, and M. Kankanhalli. Audio based event detection for multimedia surveillance. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 5, pages 813–816, May 2006.

[6] N. Ballas, B. Labbé, A. Shabou, H. Le Borgne, P.-H. Gosselin, M. Redi, B. Merialdo, H. Jégou, J. Delhumeau, R. Vieux, et al. Irim at trecvid 2012: semantic indexing and instance search. In *Proceedings of the TRECVID Workshop*, 2012.

[7] M. Bertini, A. Del Bimbo, and C. Torniai. Automatic video annotation using ontologies extended with visual information. In *Proceedings of the 13th Annual ACM International Conference on Multimedia*, pages 395–398, 2005.

[8] C. J. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998.

[9] L. Cao, S.-F. Chang, N. Codella, C. Cotton, D. Ellis, L. Gong, M. Hill, G. Hua, J. Kender, M. Merler, Y. Mu, J. R. Smith, and F. X. Yu. IBM research and Columbia University TRECVID-2012 Multimedia Event Detec-

tion (MED), Multimedia Event Recounting (MER), and Semantic Indexing (SIN) Systems. In *Proceedings of the TRECVID Workshop*, 2012.

[10] B. Caputo, H. Muller, B. Thomee, M. Villegas, R. Paredes, D. Zellhofer, H. Goeau, A. Joly, P. Bonnet, J. Martinez Gomez, I. Varea, and M. Cazorla. Imageclef 2013: The vision, the data and the open challenges. In P. Forner, H. Müller, R. Paredes, P. Rosso, and B. Stein, editors, *Information Access Evaluation. Multilinguality, Multimodality, and Visualization*, volume 8138 of *Lecture Notes in Computer Science*, pages 250–268. 2013.

[11] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.

[12] S.-F. Chang, T. Sikora, and A. Purl. Overview of the mpeg-7 standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6):688–695, June 2001.

[13] L. Chekina, D. Gutfreund, A. Kontorovich, L. Rokach, and B. Shapira. Exploiting label dependencies for improved sample complexity. *Machine Learning*, 91(1):1–42, 2013.

[14] C. Clavel, T. Ehrette, and G. Richard. Events detection for an audio-based surveillance system. In *IEEE International Conference on Multimedia and Expo (ICME)*, pages 1306–1309, July 2005.

[15] M. U. Şen and H. Erdogan. Linear classifier combination and selection using group sparse regularization and hinge loss. *Pattern Recognition Letters*, 34(3):265–274, February 2013.

[16] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 886–893, June 2005.

[17] S. Džeroski and B. Ženko. Is combining classifiers with stacking better than selecting the best one? *Machine Learning*, 54(3):255–273, March 2004.

[18] S. Ebadollahi, L. Xie, S.-F. Chang, and J. Smith. Visual event detection using multi-dimensional concept dynamics. In *IEEE International Conference on Multimedia and Expo*, pages 881–884, July 2006.

[19] J. Fan, H. Luo, Y. Gao, and R. Jain. Incorporating concept ontology for hierarchical video classification, annotation, and visualization. *IEEE Transactions on Multimedia*, 9(5):939–957, August 2007.

[20] S. fu Chang, D. Ellis, W. Jiang, K. Lee, A. Yanagawa, E. C. Loui, and J. Luo. Large-scale multimodal semantic concept detection for consumer video. In *ACM Multimedia in MIR workshop*, 2007.

[21] J.-M. Geusebroek, R. Van den Boomgaard, A. W. M. Smeulders, and H. Geerts. Color invariance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(12):1338–1350, December 2001.

[22] C. C. Gotlieb and H. E. Kreyszig. Texture descriptors based on co-occurrence matrices. *Computer Vision, Graphics, and Image Processing*, 51(1):70–86, 1990.

[23] A. Hauptmann, R. Yan, W.-H. Lin, M. Christel, and H. Wactlar. Can high-level concepts fill the semantic gap in video retrieval? a case study with broadcast news. *IEEE Transactions on Multimedia*, 9(5):958–966, August 2007.

[24] M. J. Huiskes, B. Thomee, and M. S. Lew. New trends and ideas in visual concept detection: The mir flickr retrieval evaluation initiative. In *Proceedings of the International Conference on Multimedia Information Retrieval*, pages 527–536, 2010.

[25] W. Jiang, S.-F. Chang, and A. Loui. Context-based concept fusion with boosted conditional random fields. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages 949–952, April 2007.

[26] Y.-G. Jiang, C.-W. Ngo, and J. Yang. Towards optimal bag-of-features for object categorization and semantic video retrieval. In *Proceedings of the 6th ACM International Conference on Image and Video Retrieval*, pages 494–501, 2007.

[27] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, November 2004.

[28] O. Madani, M. Georg, and D. Ross. On using nearly-independent feature families for high precision and confidence. *Machine Learning*, 92(2-3):457–477, 2013.

[29] A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen. Acoustic event detection in real life recordings. In *18th European Signal Processing Conference*, pages 1267–1271, 2010.

[30] M. Naphade, A. Natsev, C.-Y. Lin, and J. Smith. Multi-granular detection of regional semantic concepts [video annotation]. In *IEEE International Conference on Multimedia and Expo (ICME)*, volume 1, pages 109–112 Vol.1, June 2004.

[31] U. Niaz, M. Redi, C. Tanase, and B. Merialdo. EURECOM at TrecVid 2012: The light semantic indexing task. In *Proceedings of the TRECVID Workshop*, 2012.

[32] S. Nowak and P. Dunker. Overview of the clef 2009 large-scale visual concept detection and annotation task. In *Multilingual Information Access Evaluation II- Multimedia Experiments*, volume 6242 of *Lecture Notes in Computer Science*, pages 94–109. 2010.

[33] P. Over, G. Awad, W. Kraaij, and A. F. Smeaton. Trecvid 2006 – an overview. In *Proceedings of the TRECVID Workshop*. NIST, USA, 2006.

[34] E. Ozan, S. Tankiz, B. Acar, and T. Ciloglu. Content based event retrieval on tv broadcast audio. In *IEEE 19th Conference on Signal Processing and Communications Applications (SIU)*, pages 391–394, April 2011.

[35] M. Ozay and F. Yarman Vural. A new decision fusion technique for image classification. In *IEEE International Conference on Image Processing (ICIP)*, pages 2189–2192, November 2009.

[36] M. Ozay and F. Yarman Vural. A new fuzzy stacked generalization technique and analysis of its performance. *arXiv preprint arXiv:1204.0171*, 2012.

[37] S. Petridis, T. Giannakopoulos, and S. Perantonis. A multi-class method for detecting audio events in news broadcasts. In *Artificial Intelligence: Theories, Models and Applications*, volume 6040 of *Lecture Notes in Computer Science*, pages 399–404. 2010.

[38] S. Pfeiffer, S. Fischer, and W. Effelsberg. Automatic audio content analysis. In *Proceedings of the 4th ACM International Conference on Multimedia*, pages 21–30, 1996.

[39] J. Portelo, M. Bugalho, I. Trancoso, J. Neto, A. Abad, and A. Serralheiro. Non-speech audio event detection. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1973–1976, April 2009.

[40] G.-J. Qi, X.-S. Hua, Y. Rui, J. Tang, T. Mei, M. Wang, and H.-J. Zhang. Correlative multilabel video annotation with temporal kernels. *ACM Transactions on Multimedia Computing, Communications and Applications*, 5(1):3:1–3:27, October 2008.

[41] M. Robnik-Sikonja and I. Kononenko. An adaptation of relief for attribute estimation in regression. In *Proceedings of the Fourteenth International Conference on Machine Learning*, pages 296–304, 1997.

[42] B. Safadi, N. Derbas, A. Hamadi, F. Thollard, G. Quénot, J. Delhumeau, H. Jégou, T. Gehrig, H. K. Ekenel, and R. Stifelhagen. Quaero at TRECVID 2012: Semantic indexing. In *Proceedings of the TRECVID Workshop*, 2012.

[43] P. Salembier and T. Sikora. *Introduction to MPEG-7: Multimedia Content Description Interface*. 2002.

[44] N. Sebe and M. Lew. Texture features for content-based retrieval. In *Principles of Visual Information Retrieval*, Advances in Pattern Recognition, pages 51–85. 2001.

[45] Y. Shiraishi and K. Fukumizu. Statistical approaches to combining binary classifiers for multi-class classification. *Neurocomputing*, 74(5):680–688, February 2011.

[46] M. Sjöberg, S. Ishikawa, M. Koskela, J. Laaksonen, and E. Oja. PicSOM Experiments in TRECVID 2012. In *Proceedings of the TRECVID Workshop*, 2012.

[47] A. Smeaton, P. Over, and W. Kraaij. High-level feature detection from video in trecvid: A 5-year retrospective of achievements. In A. Divakaran, editor, *Multimedia Content Analysis*, Signals and Communication Technology, pages 1–24. 2009.

[48] C. Snoek, M. Worring, D. Koelma, and A. Smeulders. A Learned Lexicon-Driven Paradigm for Interactive Video Retrieval. *IEEE Transactions on Multimedia*, 9(2):280–292, February 2007.

[49] C. G. M. Snoek, S. Member, M. Worring, J. mark Geusebroek, D. C. Koelma, F. J. Seinstra, and A. W. M. Smeulders. The semantic pathfinder: Using an authoring metaphor for generic multimedia indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28:1678–1689, 2006.

[50] C. G. M. Snoek, K. E. A. van de Sande, O. de Rooij, B. Huurnink, E. Gavves, D. Odijk, M. de Rijke, T. Gevers, M. Worring, D. C. Koelma, and A. W. M. Smeulders. The mediamill trecvid 2010 semantic video search engine. In *Proceedings of the TRECVID Workshop*, 2010.

[51] C. G. M. Snoek, J. C. van Gemert, T. Gevers, B. Huurnink, D. C. Koelma, M. van Liempt, O. de Rooij, K. E. A. van de Sande, F. J. Seinstra, A. W. M. Smeulders, A. H. C. Thean, C. J. Veenman, and M. Worring. The MediaMill TRECVID 2006 semantic video search engine. In *Proceedings of the TRECVID Workshop*, November 2006.

[52] 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, January 2005.

[53] C. G. M. Snoek and M. Worring. Concept-based video retrieval. *Foundations and Trends in Information Retrieval*, 2(4):215–322, April 2009.

[54] C. G. M. Snoek, M. Worring, and A. Hauptmann. Detection of tv news monologues by style analysis. In *IEEE International Conference on Multimedia and Expo (ICME)*, volume 2, pages 1103–1106, June 2004.

[55] C. G. M. Snoek, M. Worring, and A. W. M. Smeulders. Early versus late fusion in semantic video analysis. In *Proceedings of the 13th Annual ACM International Conference on Multimedia*, pages 399–402, 2005.

[56] M. Soysal, K. B. Loğoğlu, M. Tekin, E. Esen, A. Saracoğlu, B. O. Acar, E. C. Ozan, T. Ateş, H. Sevimli, M. Sevinç, İlkay Atıl, S. Özkan, M. A. Arabacı, S. Tankız, T. Karadeniz, D. Önür, S. Selçuk, A. Alatan, and T. Çiloğlu. Multimodal concept detection in broadcast media: Kavtan. *Multimedia Tools and Applications*, pages 1–46, 2013.

[57] E. Spyrou and Y. Avrithis. Detection of high-level concepts in multimedia. In *Encyclopedia of Multimedia*, pages 151–158. 2008.

[58] M. A. Stricker and M. Orengo. Similarity of color images. In *Storage and Retrieval for Image and Video Databases III*, pages 381–392, 1995.

[59] M. Swain and D. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[60] M. Szummer and R. W. Picard. Indoor-outdoor image classification. pages 42–51, 1998.

[61] M. Tahir, F. Yan, M. Barnard, M. Awais, K. Mikolajczyk, and J. Kittler. The university of surrey visual concept detection system at imageclef@icpr: Working notes. In *20th International Conference on Pattern Recognition (ICPR)*, pages 850–853, August 2010.

[62] X. Tan, S. Chen, Z.-H. Zhou, and F. Zhang. Recognizing partially occluded, expression variant faces from single training image per person with som and soft k-nn ensemble. *IEEE Transactions on Neural Networks*, 16(4):875–886, July 2005.

[63] K. M. Ting and I. H. Witten. Issues in stacked generalization. *Journal of Artificial Intelligence Research*, 10:271–289, 1999.

[64] L. Todorovski and S. Džeroski. Combining classifiers with meta decision trees. *Machine Learning*, 50(3):223–249, March 2003.

[65] B. Tseng, C.-Y. Lin, M. Naphade, A. Natsev, and J. Smith. Normalized classifier fusion for semantic visual concept detection. In *Proceedings of the International Conference on Image Processing (ICIP)*, volume 2, pages II–535–8 vol.3, September 2003.

[66] N. Ueda. Optimal linear combination of neural networks for improving classification performance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(2):207–215, February 2000.

[67] K. van de Sande, T. Gevers, and A. Smeulders. The university of amsterdam's concept detection system at imageclef 2009. In *Multilingual Information Access Evaluation II- Multimedia Experiments*, volume 6242 of *Lecture Notes in Computer Science*, pages 261–268. 2010.

[68] J. C. van Gemert, J.-M. Geusebroek, C. J. Veenman, C. G. M. Snoek, and A. W. M. Smeulders. Robust scene categorization by learning image statistics in context. In *Proceedings of the Conference on Computer Vision and Pattern Recognition Workshop*, pages 105–, 2006.

[69] M. Viola, M. J. Jones, and P. Viola. Fast multi-view face detection. In *Proceedings of Computer Vision and Pattern Recognition*, 2003.

[70] D. Wang, X. Liu, L. Luo, J. Li, and B. Zhang. Video diver: Generic video indexing with diverse features. In *Proceedings of the International Workshop on Workshop on Multimedia Information Retrieval*, pages 61–70, 2007.

[71] F. Wang, Z. Sun, D. Zhang, and C.-W. Ngo. Semantic indexing and multimedia event detection: ECNU at TRECVID 2012. In *Proceedings of the TRECVID Workshop*, 2012.

[72] Y. Wang, Z. Liu, and J.-C. Huang. Multimedia content analysis-using both audio and visual clues. *IEEE Signal Processing Magazine*, 17(6):12–36, November 2000.

[73] P. Wilkins, T. Adamek, N. O'Connor, and A. Smeaton. Inexpensive fusion methods for enhancing feature detection. In *International Workshop on Content-Based Multimedia Indexing (CBMI)*, pages 114–121, June 2007.

[74] D. H. Wolpert. Original contribution: Stacked generalization. *Neural Networks*, 5(2):241–259, February 1992.

[75] P. Wu, B. S. Manjunanth, S. Newsam, and H. Shin. A texture descriptor for image retrieval and browsing. In *Proceedings of the IEEE Workshop on Content-Based Access of Image and Video Libraries (CBAIVL)*, pages 3–7, 1999.

[76] J. Yang, Y.-G. Jiang, A. G. Hauptmann, and C.-W. Ngo. Evaluating bag-of-visual-words representations in scene classification. In *Proceedings of the International Workshop on Multimedia Information Retrieval*, pages 197–206, 2007.

[77] M. yu Chen and A. Hauptmann. Multi-modal classification in digital news libraries. In *Proceedings of the 2004 Joint ACM/IEEE Conference on Digital Libraries*, pages 212–213, June 2004.

[78] J. Yuan, H. Wang, L. Xiao, W. Zheng, J. Li, F. Lin, and B. Zhang. A formal study of shot boundary detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(2):168–186, February 2007.

[79] J. Yuan, H. Wang, L. Xiao, W. Zheng, J. Li, F. Lin, and B. Zhang. A formal study of shot boundary detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(2):168–186, February 2007.