Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 45 No. 2 (2024)

A Review on Feature Combination in Object Classification

Amitav Mahapatra [1], Prashanta Kumar Patra [2]
[1]*Biju Patnaik University of Technology, Rourkela, Odisha, India*
[2]*SOA University, Bhubaneswar, Odisha, India*

Abstract:- Research and development could focus on feature combination. It is used in metric learning, text classification, scene classification, disease detection, object classification, and more. Even while SIFT, HOG, and SURF are powerful with some classes, they cannot handle all classes. Inter-class correlation and intra-class diversity make object classification challenging. Using feature combination to combine the strength of numerous complementing features improves multiclass object classification results that are robust to intra-class changes and interclass correlations.

Baseline, boosting, MKL, and other feature combining approaches exist. Even though researching and trying have yielded many important evaluations on these methodologies, more review is needed.

Keywords*:* Boosting, Baseline, MKL.

1. Introduction

The boundless expansion in the field of computer vision provides a broader area for immense research activities and innovation. Even with simple image data sets, designing a successful object categorization system is difficult, even though the computer vision domain has focused on it for decades. Some images within a class differ greatly while others within different classes are similar due to intra-class variability and inter-class correlation.

Some important tools to detect and describe single features like SIFT [30], HOG [31], SURF [32], and have been proposed to solve the above problem, but they have a major drawback: they are powerful with some object classes but not enough to support all. The strength of multiple complementary features must be combined to create a final feature with superior classification accuracy than any single one.

In this introduction part, we have given a brief about feature combination, its purpose and methods.

1.1 Single features

Computer vision's object recognition and classification discipline is growing and exciting. To simplify item detection and description, single features are introduced:

SIFT

SIFT [30] extracts distinct and invariant characteristics from images for valid object matching. It mainly finds and describes local image features. Features match well across affine distortion, noise, illumination, and 3D perspective shifts and is invariant to picture scale and rotation. Its applications include 3D modeling, motion tracking, robotic mapping and other.

HOG

HOG is another common computer vision and image processing feature descriptor for object detection [31]. In limited picture regions, it counts gradient orientations. It computes on a dense grid of regularly spaced cells and uses overlapping local contrast normalization for accuracy, unlike SIFT.

HOG/SIFT has many meanings. Edge or gradient structure, typical of local shape, is acquired in a local representation with easily adjustable variance to local geometric and photometric modifications.

SURF

SURF (SPEEDED UP ROBUST FEATURES) [32] is a fast and accurate interest point detector-descriptor that outperforms other tough descriptors. It was influenced by SIFT. SURF's standard version is faster and more resilient to image transformations than SIFT.

Apart from these aforesaid excellent performing single features, there exist various other feature detectors and descriptors on which a lot of research work has been carried out. GLOH is another variant of SIFT that takes into consideration more spatial regions for the histograms. It uses a scheme called PCA to reduce higher dimensionality of the descriptor and is proved to be more distinctive with the same number of dimensions. However it is computationally quite expensive. Although various refinements to the basic SIFT descriptor have been proposed, it still seems to be the most fascinating descriptor for practical uses, hence most widely used. In some papers like [26], a new descriptor is proposed by combining three popular local descriptors i.e. SIFT, LBP[33] and GBPWHGO [17].The new descriptor bears more information than single one and can effectively handle variance of scale. Some frequently used detectors in image processing include canny detector that detects edges, FAST (Feature from accelerated segment test) which detects corners, Grey-level blobs detects blobs and others.

Some descriptors like ORB are termed as binary desriptor.ORB stands for Oriented FAST and rotated BRIEF. It combines FAST and BRIEF with several tweaks to improve performance. ORB outperforms SIFT and SURF in computing cost, matching performance, and patents. The patented SIFT and SURF need payment to use. ORB isn't patented, unlike OpenCV.

1.2 Feature combination overview

As said before, feature combination combines complementing traits to create a stronger one. Feature combination involves weighting and/or selection. The formal definition from [2] is repeated here

Feature combination problem (definition):

$Given\ a\ training\ set\ \{(x_i, y_i)\}_{i=1,\ldots N}\ \ of\ N\ instances\ consisting\ of\ an\ image\ x_i\ \in$
$X\ and\ a\ class\ label\ y_i\ \in \{1, \ldots C\}, and\ given\ a\ set\ of\ F\ image\ features\ f_m\ : X\ \rightarrow R^{\,d_m}, m =$
$1, \ldots F\ \ where\ d_m\ denotes\ the\ dimensionality\ of\ the\ m'th\ feature,$
$the\ problem\ of\ learning\ a\ classification\ function\ y : X \rightarrow \{1, \ldots C\} from\ the\ features$
$and\ training\ set\ is\ called$ feature combination problem.

1.2.1 Methods of feature combination

Several techniques solve the feature combination problem. Kernel methods predominantly use kernel functions to learn multiclass classifiers from training data. Kernels can be associated with image features.

Feature selection as kernel selection:

As a kernel is associated with a feature, implies that kernel selection/combination transforms to feature selection/combination. Cross Validation (CV) is an approach to find a kernel from the set $\{k_1, \ldots k_F\}$.

The following are the classes of Feature combination in brief [2]:

1. BASELINES

Deterministically combining kernels for SVM training is covered by two baseline techniques. These are:

Average Method

The easiest approach to merge kernels is average. Definition of kernel function is

$$k^*(x, x') = \frac{1}{F} \sum_{m=1}^{F} k_m(x, x') \ .$$

While several combination algorithms have been proposed, this baseline method is worth to be compared with.

Product method

As the name reflects, this baseline method fuses several kernels by multiplication. Here

$$k^*(x,x') = \left(\prod_{m=1}^{F} k_m(x,x')\right)^{1/F}$$

is the kernel function definition that is used as the single kernel in a SVM.

2. MKL (multiple kernel learning)

This kernel selection method uses known kernels rather than producing fresh ones and discovers an optimal linear and non-linear combination during algorithm training. MKL optimizes linear kernel combinations simultaneously.

$$k^*(x,x') = \sum_{m=1}^{F} \beta_m\, k_m(x,x')$$

and the parameters $a \in R^N$ and $b \in R$ of an SVM. There are strong reasons behind the rigorous use of multiple kernel learning which include a) it provides a larger set of kernels for the selection of kernels and parameters and b) combines data from distinguished sources (images, sound etc)

3. BOOSTING

Boosting method gathers features in a mixed way that are coded by dissimilar systems. Inspired by the MKL decision function, boosting method is based on adding new kernels iteratively until a stopping criterion is reached. To improve weak learning algorithms, boosting was invented. An updated boosting method, Ada-Boost, combines weak classification functions to strengthen weak classifiers.

The intuitive idea that 'more features lead to higher performance' boosted feature combination research for principled classification systems. Some old feature combination approaches and recent algorithm design methods have been reviewed and adjusted to find the optimal algorithm.

SVM classifiers transform feature combinations to kernel combinations. MKL (multiple kernel learning) is a prominent kernel combination method that optimizes kernel weights to provide the greatest combination performance.

$$k^*(x,y) = \sum_{i=1}^{n} w_i\, k_i(x,y)$$

alongside SVM parameters a and b [6]-[10]. Canonical MKL assumes a consistent weighting scheme across the input space, while sample-specific MKL based kernel weights on kernel functions and samples. The former technique boosts performance but increases computation and overfitting. The two contradictory approaches were addressed using group-sensitive MKL.

MKL has been eye-catching recently, but not always. It has no advantage over baseline average combination sometimes. Although some MKL-like kernel combination algorithms have been authorized, these optimization methods are controversial. Optimization algorithms waste memory and are expensive to compute. MKL may have been exaggerated due to lack of comparison with simple but powerful baseline approaches..

Instead of optimizing all parameters at once, [2] suggested using LPBoost to train them independently in two phases. Boosting approaches consistently outperform MKL and baselines.

Traditional techniques of feature combination Conventional methods for combining features involve (1) merging various feature vectors into a single composite vector and training a classifier on it (Dassigi et al., 2001); (2) forcefully combining all the features into a single long feature vector and reducing its dimensionality using techniques such as PCA (Principal Component Analysis), LDA, etc. The initial two techniques are referred to as "Feature Fusion," while the final one is known as "Decision Fusion."These two strategies dominate "Data Fusion". The aforementioned feature combination strategies can be split into three tiers (low, middle, and high) for effective classification from human perception. With low-level feature combination, features are worked on without refining. The middle level selects a subset of original features. The higher level of feature combination combines classifiers from different systems' programmed feature sets. However, each level has flaws. For low-level feature combinations, worthless, redundant, or incompatible features may cause poor classifier performance on lengthy feature vectors. Sometimes the composite vector has too many dimensions, making computation expensive. Even with advanced feature selection approaches, the middle level may not contain an ideal feature set. The high level of feature combination may lack feature communication in training because it focuses on component classifiers.

1.2.2 Different approaches to Feature combination

In other words as described in [27], features can be combined through one of the following processes stated below.

Early combination: Better known as feature level combination. Here, the features are aggregated before applying any classifier. The initial step involves extracting features from the datasets. These features are then concatenated to generate a new feature vector. This new feature vector is subsequently utilized to train the classifier. Its applications include that of human and face recognition. In some methods feature concatenation is performed after dimensionality reduction while others implement the same before feature selection and transformation. Because of its simplicity, this method is commonly used. The block diagram can be viewed in Fig. 1

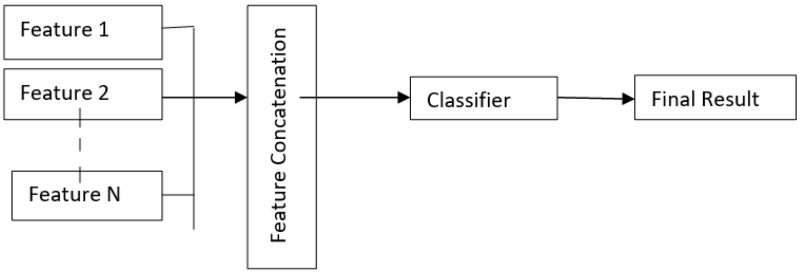

Fig. 1 Early combination

Intermediate combination: Figure 2 shows how mapping from feature space to decision space combines features. In intermediate combination, kernel values are evaluated independently for each descriptor and merged to create a new kernel matrix for the SVM classifier. Linear kernel combinations are suitable examples. The idea is new because most research focuses on generating a weighted mixture of kernels that is optimal. Validation sets are a simple way to discover the optimal weighted kernel combination for performance.

A drawback of this approach is that it’s difficult to keep track when the number of kernels accelerates. Thus optimization technique like MKL was considered which could solve a joint optimization problem as well as learn the optimal weights for combining the kernels.

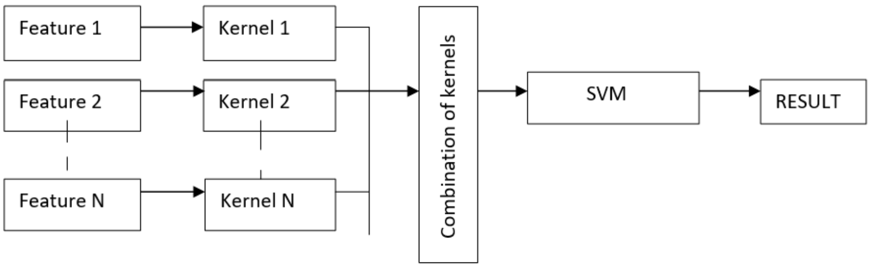

Fig. 2 Intermediate combination

Late combination

Here features are combined after the mapping takes place from feature space to decision space. So the other name of this method is decision combination. For each feature vector, a classifier is trained and makes a decision. The outputs of the classifier are subsequently combined using methods such as weighted summation, weighted product, majority voting, logic operators such as AND and OR, or sorted lists. The use of late combining has gained popularity and proven to be more effective than alternative image classification methods. Fig. 3 shows this method's block diagram.

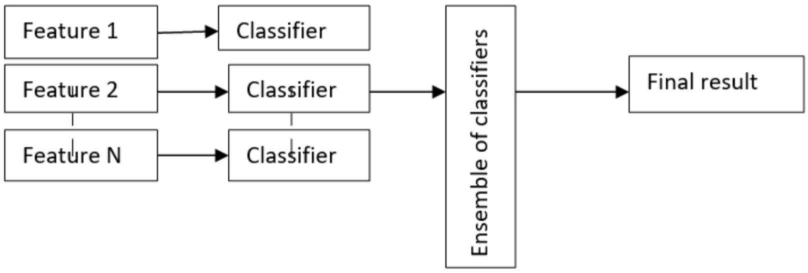

Fig. 3 Late combination

2. Literature Works

Although feature combination work is extended to different branches like text classification, disease diagnosis, human faces detection etc. here we stick to the works on the Feature combination in 'Object classification'. We elaborately present the literature works carried out in the past, their methods adopted, comparisons, advantages and disadvantages of certain methods involved and finally a summary. So the literature works are as follows:

A new boosting-based feature combination method is proposed [1]. Boosting combines features here. Unlike ordinary boosting, variation boosting trains weak classifiers on multiple feature sets. Finally, weighted voting combines classifiers. Output classifier for that round. Research shows this feature combination technique can combine feature selection, communication, and classifier learning. Schapire (1990) introduced boosting to improve weak learning algorithms. AdaBoost extends boosting and voting classifiers through two classes. A similar feature vector is given to weak classifier components in conventional AdaBoost. Even though all features are different types, each training instance includes a fixed-length feature vector with similar attributes in a specified order. As seen in [1], AdaBoost works better. This boosting strategy trains a middle final classifier from several weak classifiers on samples of each feature vector using system-coded characteristics at each cycle. Each round of this algorithm identifies combinations using weighted voting. Weak learning algorithms can be chosen for AdaBoost and other boosting algorithms like decision trees, neural networks, and others. This boosting variation can be changed in multi-class cases like general boosting. The suggested method in [1] classifies specifically, unlike AdaBoost, which classifies broadly. The variant of boosting showed much better classification performance than traditional approaches on three data sets. Boosting, feature extraction optimization, and neural network parameter adjustment can improve its performance.

Further works on boosting in [2] confirmed its consistency and robustness by extending it to several multiclass setting. In [2] vivid kernel methods of feature combination have been defined in a clear manner which includes baselines, MKL and boosting. The paper proposed formulations based on LPBoost. In particular two methods have been proposed that are inspired by the MKL decision function. They are LPBoost and its multiclass variants i.e. LP-β and LP-B that have been experimented on specific datasets. In MKL solution, combining coefficients are regarded as feature influence on a class, but in multiclass situation, this is false. Thus, multiclass decision-making values all features equally. LP-β chooses three of seven features, while other approaches aim to choose all features. Oxford flower experiment results show MKL and LP-β learning methods are resilient to unnecessary characteristics, but CG-Boosting strategy worsens with time.

A new feature combination technique based on boosing is proposed in [1]. The initial experiment (oxford flower) shows that MKL with boosting can select meaningful kernels from a broad class of possibly uninformative ones. Performance won't depend on kernel selection if each feature is discriminative. Other observations were made after experimenting on Caltech101 and Caltech256 datasets. The baseline approaches surpass SIFT features MKL and CGBoost's pyramid kernels and perform worse than a single kernel.

PHOG pyramid kernels underperform CGBoost and MKL.The LP-β (Boosting) method provides best outcomes for both combinations. Therefore, LPBoosting performs best while Caltech 101 MKL and baselines are similar. Also, LP-β has a similar runtime to MKL. Prior methods are orders of magnitude slower than baseline procedures.

Caltech256 was difficult for CGBoost. Additionally, both Caltech data sets perform 10% better than the prior best result.oosting. Boosting version is used here.

An enhanced performance is expected if more image features are trained and other classification functions are included. The two steps training associated with boosting is evidently less systematic (which is considered to be its limitation) than a joint optimization like MKL but in practical case this problem can be ignored and it works well even with few training examples. Since training comprises two phases, most of it may be done simultaneously and quickly. Most of the outcomes are bad for MKL, and the baseline has competitive and sometimes better results. It may be because the kernels are already discriminative. In the presence of uninformative kernels, boosting approaches find informative kernels, improving performance. From the amazing observations, the MKL's performance may have been overestimated in prior study. Consider average and product as its canonical opponents. With LP-β, a novel method was developed for faster, higher performance, and sparse multiclass object categorization.

Instead of monotonous time-consuming learning procedures in MKL, prior information was introduced into kernel mixing [3]. So, the weight of each attribute in combination depends on how well it classifies the class. Thus, numerous methods that combine local feature weights are presented in [3] along with the new strategy of establishing feature weights based on classification performance. Feature combination is solved with bag-of-words histogram features and kernel-based classifiers. The factor-based algorithm is

1) Each bag of words feature has a kernel.

2) Each kernel trains SVM classifiers.

3) Validation dataset predicts classifier performance.

4) Each class combines first-step kernels into one kernel and uses third-step performance ratings as coefficients.

5) The final classifier predicts validation dataset picture labels using the single kernel.

The fact that prediction ability weights each feature's effect on the final prediction makes this technique strong. To mix features and kernels, performance prior knowledge is calculated after normalising intermediate values. This historical information might be included two ways:

A. Knowledge-weighted linear kernel

B. Knowledge-Weighted Product Kernel

Knowledge weighted linear kernel can be expanded to knowledge weighted product kernel since product kernel

is naturally related to average kernel. This kernel is also fed straight into the class SVM classifier. Intensive sampling and Harris-laplace detector extract local image regions. Characterise locations with five descriptions. Opponent, C-SIFT, rgSIFT, and transformed colour SIFT. Nearest neighbour mapping maps image features to the codebook. The best feature combination method is KWPK. The results demonstrate that the proposed strategy can produce demanding results with cutting-edge methods and low technology. MKL method observations show that KWPK and KWLK outperform averaging and product kernel. On chair, bus, and other data sets, KWPK and KWLK exceed MKL, but MKL outperforms them on others. KWPK and product kernel outperform linear combination methods KWLK and average kernel. In the validation dataset, MKL performs best, KWPK and KWLK somewhat worse but comparable and competitive, and product kernel and average kernel poorly. Beyond performance, KWPK and KWLK strategies process faster than MKL. MKL spends the most time learning coefficients, while KWPK and KWLK merely train SVM classifiers and evaluate feature performance. The kernel for each feature is pre-computed and the only coefficient to train is fixed, therefore this time can be omitted. All these assessments show that KWPK and KWLK can produce results comparable to MKL while saving time. For MKL-compatible feature combination, simple and efficient methods use feature performance as previous knowledge. Future work includes answering the statistical machine learning literature's knowledge affiliation question.

Feature weight calculation is well-documented, however feature selection is rarely considered. In [4], feature selection employing several descriptors and detectors was introduced to combine powerful features to increase classification accuracy. For performance, this research examines the effect of numerous feature-related elements on feature combination, including popular descriptors, kernels, and spatial pyramid, on four data sets of distinct object categories. Thus, this article addresses kernel combination feature concerns. These descriptors include PHOG, LBP, GIST, Gabor, and RFS filter. In this work, the kernels for each feature are integrated into one feature. Evaluation proceeds as follows. First, 10 training-testing splits are employed for each classification feature, with mean recognition rates reflecting their discriminative potential. Second, features are combined in ascending, descending, and mixed modes. Features are ordered by discriminative power in each scenario. To test combination performance, features are introduced one by one.

Here are some intriguing conclusions. First, numerous powerful features increase performance more than the best single feature. The four datasets' recognition rates peak with roughly four powerful characteristics. Less powerful features don't affect performance. Second, weak features always lower recognition rates in mixed mode. Third, stronger features always increase combination performance in ascending mode until all features offer the best outcomes. To everyone's intuition, the strongest and weakest traits are complementary, thus they generate the strongest combined performance. This is not supported here because starting with the strongest features enhanced performance and then adding weaker ones lowered it. This is analogous to sparse solutions in MKL or LP-β [2], which only examine a portion of characteristics in the final combination. Unlike feature combinations, kernel combinations barely increase classification performance. Although several kernels capture different aspects of feature similarity, their average combination doesn't exceed the best single kernel. Although disappointing, this proposes a new optimization combination strategy that uses several kernel combinations to exceed baseline averaging. This study examines spatial pyramid levels. A spatial pyramid is a common feature model that uses spatial information. Steps were taken to examine pyramid levels' effect on feature combination. Despite higher levels not improving performance, multiple levels perform better than single levels.

As noted, average and product kernel combinations are the easiest. Average combination technique features have been extensively studied in [5] and shown to be compatible with the kNN architecture. Based on this kNN paradigm, selection-based average combination (SBAC) was recommended over basic average combination. Based on [4] experimental arrangements, [5] extended its critical observations using kNN framework and the novel method. The graph encounters a 'rise-peak-drop' phenomenon in the descending mode of feature addition into the combination, therefore guessing the k in kNN is the only goal. Cross Validation measures the discriminative strength of characteristics and finds it effective; hence it is used to arrange them. By considering all features, SBAC outperforms the usual average combination approach. The performance of Event-8, Flower-17, Scene-15, and Caltech-101 is noteworthy. In [2], the MKL approach outperforms the regular average combination method by orders of magnitude, similar to the performance increases in this paper[5].This motivates revisiting the average combination approach since it emphasizes the best results so far and encourages greater effort to create and improve the baseline method. The performance improvement varies with data set size, which is intriguing. Flower-17 and Caltech-101 have a substantial performance boost, while Scene-15 and Event-8 have a smaller gain. The k-nearest-neighborhood paradigm supports the discriminative power of individual features that explain such disparities. The observations above demonstrate that the kNN framework is good at understanding feature combination behavior. SBAC is simple, powerful, and a feature combining baseline. In addition, the kNN design demonstrates how features improve performance. All of these results suggest that the kNN framework stimulates unique feature combination algorithms by focusing on feature combination process comprehension. MKL, a more advanced approach [6, 7, 8, 9], has garnered attention, but its efficacy is inconsistent. Advanced research in [10] proposed a non-linear kernel combination to boost performance.

Other non-linear kernel learning works include [11, 12]. The sequential minimum optimisation (SMO) approach was introduced to train -norm MKL in [13]. The SMO technique accelerated massive data and kernel space training. When MKL-like approaches are used, adding kernels can have a big impact [14]. MKL's resilience was further examined in [15], where SSaC (soft salient coding) was proposed to minimise the information suppression problem in the original SaC approach. In the spatial pyramid matching (SPM) framework, the SSaC feature was analysed with two additional features and MKL approach was used for classification. The results showed that the features work together to improve performance. The suggested MKL feature combination schematic is below.

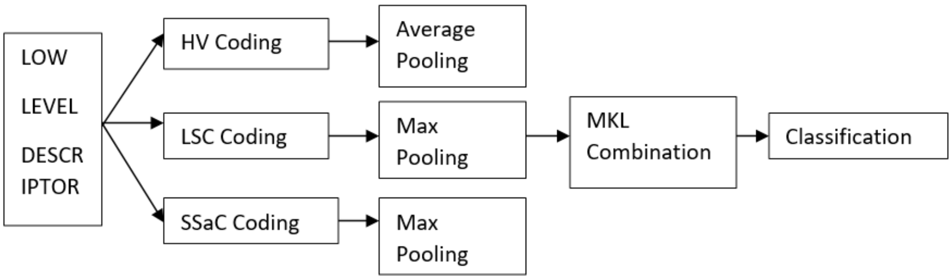

Fig. 4 Proposed MKL method

For classification, MKL integrates distinct picture characteristics derived from the same low-level descriptors, inspired by SPM's encoding and pooling stages, where different algorithms captured different image qualities [15]. To improve image classification, the MKL may incorporate picture attributes from several encoding and pooling algorithms. Here, HV, LSC, SaC, and average and max pooling are employed. A harsh voting procedure like SaC can suppress a lot of visual information when max pooling pools features. Soft SaC was implemented to remedy this. Three actual contributions from this study:

1. To prevent information suppression in the original SaC approach, the SSaC method has been presented, revealing that more picture information can be used for classification. GSaC (group code size-based SaC) has been considered to test SSaC.
2. The meaning and features of code created by different encoding and pooling algorithms have been investigated.Image characteristics vary by encoding and pooling method.
3. MKL can adjust picture features from multiple classes, therefore kernel approaches are used for classification. Additional performance effects are examined when MKL is regularized using and norm. The effectiveness of these major ideas is tested on three data sets.

The suggested SSaC strategy was tested using Caltech-101 datasets. The k-Means algorithm is utilized. Even with HOG descriptors, SSaC performs best. SSaC operates more reliably than GSaC and SSaC. SSaC and GSaC perform worse with larger step sizes, although SSaC decreases slower. The three techniques perform well on HOG descriptors, but SSaC stands out. Regardless of the number of training samples, the SSaC performs best with distinct descriptors and training samples from each category.As seen, MKL with different regularizations performs differently. MKL with norm performs worse than separate coding schemes under constrained characteristics. However, more training data narrows the performance gap. Best MKL performance occurs with more training data per category. We need enough training data for MKL with norm to perform better. Unlike the above results, MKL with norm performs substantially better. MKL with virtually always yields the greatest results, and the trend becomes evident as training images rise. This shows that several features work together to increase performance. It is evident that sparse weight responses for distinct features under different regularizers result in different kernels being picked for different classes. In contrast, MKL with will coordinate and integrate feature significance for categorization. The proposed technique is correct because regularizer-selected features consistently yield big values when regularized by norm.

Redundancy aids feature combination. A new feature processing method that integrates original features with redundancy cutting is shown in [16]. This method outperforms typical classification methods. According to feature type, the experiment proposes multiple critical feature processing processes. Steps include numerical to categorical feature value conversion, feature combination, redundancy discrimination, and latent structure finding with original and extended features. For demonstration, UCI repository is used. Initial results indicate that the suggested method outperforms SVM in classification accuracy while ROC benchmark remains equal to SVM. The discrete topic vector model completes feature processing. Two assumptions underpin this simple feature processing flow method. First, measurement errors may affect data record feature values. i.e., real-world data features may have been measured by some techniques that produced some measurement errors. Second, because data mining jobs use distinct characteristics, the feature data obtained may be redundant.

The suggested strategy emphasizes feature combining to generate a new, enlarged feature collection. To accomplish the result, several actions are followed in order. To categorize numerical features, k-means clustering is used. We found that tf-idf threshold of 0.04 yields excellent classification results: accuracy near 55% and ROC 0.702. The accuracy is 54.5607% and ROC is 0.702 compared to the typical SVM experiment on the same data set. This shows that the proposed feature processing flow can improve ROC benchmark accuracy.

A combinational strategy to develop extended features to speed classification performance has been shown to be promising in field feature processing. The proposed technique improves classification performance, although more optimization is needed. Future work is needed in these areas:

1. The translation of numerical to category features significantly impacts later processing steps. As feature combination is based on conversion output, choosing the clustering result correctly affects the combination result. Each feature should have a distinct cluster number. Thus, setting cluster numbers for each characteristic is crucial for the future.
2. The degree of combination determines the number of created extended characteristics. Though the combination process can yield significant patterns, a higher degree of combination will rapidly increase the number of extended features. Effective degree parameter selection affects time and spatial complexity.

Low-level or semantic modelling was used in most scene categorization algorithms. Both approaches are flawed. The low-level technique classifies pictures into few scenes and performs poorly.

High computational and memory costs plague the semantic technique. In [17], a revolutionary strategy retains the benefits of both strategies while overcoming their drawbacks. To represent scene photos, GBPWHGO is used. Direct image representation is possible with the GBPWHGO descriptor. The GBPWHGO descriptor efficiently captures picture structural and textural attributes using a GBP and WHGO.

The experimental setup is centered at GBPWHGO with some peripheral descriptors. In comparison to the “Spatial Pyramid Matching” of [18], it is found that:

1. The SPM approach extracts hundreds or thousands of local characteristics per image, while the suggested method only requires 28.
2. This approach is straightforward and computationally efficient, as it avoids the need to quantify a large number of local descriptors into a codebook.

The results demonstrate that SPM method derived features use more memory than the currently recommended method.Many feature combination studies have been published in recent decades. Many issues remain unresolved. Gehler and Nowozin found in [2] that MKL do not outperform the baseline average combination when all kernels are potent.

The optimization-based MKL technique only achieves superior performance when it combines strong and weak characteristics, resulting in the reduction of weak features.

Due to insufficient baseline comparisons, MKL capabilities may have been exaggerated. The popular MKL approach gains little performance over the typical baseline method but consumes more compute and memory. This prompted researchers to reconsider the average combination, which they considered inferior. Hou and Pellilo devised the dominant set clustering technique [21] to generate discriminative power and kernel weights. This was done by investigating the relationship between the SVM classification mechanism and dominant set clustering.

Dominant sets added a new dimension to feature combination. Simply put, dominant sets are clusters. The graph-theoretic concept of a cluster does not necessitate prior knowledge of the quantity of clusters.It handles asymmetric and negative similarity functions from pair-wise similarity matrices. A cluster is a set of data that are very homogeneous within and heterogeneous outside. It's fascinating how sets determine the amount of clusters. Dsets clustering can be done by peeling apart clusters from input data [22].According to [21], multiple datasets were utilized to assess a kernel matrix's accuracy for an SVM classifier. The anticipated accuracy of the kernel matrix reflects its capacity to discriminate and is used to assign weight to the kernel matrix in feature combination. Similar to [21], [23] employed dominant sets to assess accuracy using kNN architecture. The classification accuracy of [21] and [23] is higher than other approaches.

With four data sets—Caltech 101, scene -15, event-8, Flower-17—the weighted combination approach of [21] enhances event-8 and scene 15 results significantly. Average combination outperforms greatest individual attribute. It also outperforms CV and MKL on both data sets, confirming its efficacy. The results show that weighting has little advantage over average combination when feature performance variance is modest. According to [2], existing learning approaches have a little advantage over average combination if all kernels have the same discriminative strength. Weighting improves performance for Caltech-101 and Flower-17, the other two data sets. Additional tests demonstrate that the proposed weighting technique, similar to MKL and LP-β, effectively reduces the negative effects of weak characteristics when used in combination. The LP-β algorithm demonstrates superior performance compared to the weighting scheme in these data sets, however it is still superior to LP-average and MKL. This implies that by formulating it correctly, this straightforward and intuitive methodology for weighting kernel correctness can be just as effective as complex optimisation methods. This methodology can serve as a benchmark combination method, similar to the average combination, because of its strong performance, capacity to adjust to computational complexity, and efficient use of memory. Aside from classification accuracy, the computational cost is also a crucial factor in feature combination algorithms, particularly when multiple attributes are involved. Based on the analysis of the running time of various approaches, it can be concluded that the suggested weighting method is significantly more efficient than CV and MKL. It primarily focuses on clustering dominant sets in each kernel matrix during its activity. In this paper, feature weighting follows this flowchart.

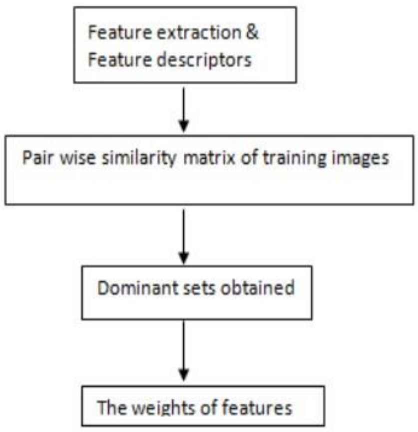

Fig. 5 Feature weighting flowchart

Features weights are extended to kNN framework in [23].Average combination feature behaviors can be seamlessly integrated into instance-based learning's k-nearest neighbor architecture. Noise can reduce classification accuracy, therefore k=1 is frequently not the best choice. Multiple nearest neighbors improve categorization accuracy as k increases. If k grows greatly, accuracy decreases. Therefore, its classification accuracy rises and falls. DSets clustering was used to create a kernel combination technique in [24], unlike MKL optimization approaches. This paper's technique is classifier-independent. The clustering methodology employed in this study involves the following steps: graph generation, DSets clustering, selection of the target cluster, and kernel weighting. The steps involved include assessing the compatibility of two kernels for combination, creating the kernel graph, performing clustering, and selecting the desired cluster.Finally, the target cluster base kernel weights are optimized. Kernel combination aims to maximize classification accuracy. Since cross validation accuracy estimates the classification capacity of the combined kernel, kernel combination aims to weight base kernels effectively to optimize cross validation accuracy. High cross validation accuracy makes two base kernels appropriate for combination. This is two kernel combination suitableness. A graph with weighted edges is generated by combining the suitability of two base kernels as adjacency. As proven in [24], DSet clustering method may solve combination suitableness maximization given the pair-wise kernel adjacency matrix.This is just kernel combination turned into clustering.The primary goal of this paper is to maximise the cross validation of the combined kernel, which is not influenced by the SVM classification process. In addition, the optimization method only uses the pair-wise kernel adjacency matrix, requiring less memory and compute than MKL algorithms. This study primarily concentrates on the categorization of pictures. However, the technique of combining features can also be applied to tasks such as pattern classification, defect diagnostics, and so on. Given that [21] use the DSets technique in kernel combination, the distinctions can be outlined as follows. Firstly, [21] posits that discriminative kernels should be given significant weights when combined, and that the weight of each kernel should be computed independently. In contrast, [24] only includes appropriate kernels and decides the weights of the kernels based on their suitability for combination. Furthermore, in reference [21], the kernel matrix is individually inputted into DSets clustering, which calculates the discriminative power and weight of each kernel. DSet clustering assesses the discriminative strength of the kernel, substituting the need for cross validation. On the other hand, DSets clustering identifies the appropriate base kernel subset to merge and assign weights to [24]. Furthermore, in reference [24], the DSets clustering algorithm utilises a kernel adjacency matrix and clusters all base kernels, which is a distinction from reference [21]. Each kernel matrix is processed separately using DSets to extract all possible clusters [21]. However, in reference [24], the DSet approach is specifically used on a single kernel adjacency matrix, resulting in the extraction of only the initial cluster. The latter strategy is more straightforward and grounded in a more rational concept than the former.

Therefore, the clustering-based technique is efficient since it utilises two kernels and assesses their combination based on their performance. When the kernels exhibit excellent classification accuracy, it indicates that they are both varied and complementary. Nevertheless, the DSets technique imposes a constraint on the internal similarity of clusters, resulting in a situation where all data within the first cluster exhibit a significant degree of similarity. This suggests that any two kernels can be combined.Another source, referenced as [26], presents a kernel fusion technique that utilises LBP, SIFT, and GBPWHGO. This method has shown good outcomes. The feature combination in [27] utilises a combination of SIFT, LBP, PHOG, and GIST descriptors. The pictures were identified using the SVM classifier with a linear kernel. The data is initially modelled by hierarchical latent dirichlet allocation (hLDA) in [28], which then combines features to generate a summary. As stated in reference [29], utilising Maximal Frequent pattern-based feature selection on one-class is a highly effective mathematical approach. Feature combination is a promising topic with many innovative concepts developed and coming. Below is the literature summary.

Table 1. Summary of the literature works

TITLE	AUTHOR & YEAR	TECHNIQUES IMPLEMENTED/METHODS	FINDINGS/RESULT	FUTURE WORK/CONCLUSION
FEATURE COMBINATION USING BOOSTING.	Xu-Cheng Yin,Chang-Ping Liu,Zhi Han 2005	A variant of boosting based on AdaBoost (algorithm)	1. Decreases the test error. 2. Better classification performance 3. To some extent, it can integrate F.S, F.Communi. & classifier learning.	1. Its performance might be improved through utilizing operation on boosting, optimizing feature extraction and tuning parameters of a neural network.
ON FEATURE COMBINATION FOR MULTICLASS OBJECT CLASSIFICATION.	Peter Gehler & Sebastian Nowozin (2009)	1.Kernel method 2.Baselines 3.MKL 4.Boosting	1.10% improved performance.	1.Baseline yield competitive results & outperform MKL on several steps.

TITLE	AUTHOR & YEAR	TECHNIQUES IMPLEMENTED/METHODS	FINDINGS/RESULT	FUTURE WORK/CONCLUSION
				2.LP-β approach consistently outperforms all other methods. 3.Performance of MKL might have been over estimated in the past.
AN EFFICIENT STRATEGY FOR FEATURES COMBINATION.	Zhang,Xiao, Wang,Cheng,Shao, (2010)	1. knowledge weighted linear kernel	1 KWPK and KWLK can yield similar outcomes to MKL. 2.The processing time is highly reduced	1. This use performance as the prior knowledge for combination. 2. Simple but efficient. 3. The integration of knowledge into statistical machine learning algorithms remains an unresolved matter.
SUMMARIZATION BASED ON MULTIPLE FEATURE COMBINATION	Huang,Li, Zhang,Chi(2010)	hLDA modelling		Effective in extracting summaries.
EVALUATING FEATURE COMBINATION IN OBJECT CLASSIFICATION.	Hou,Zhang,Qi,Yang (2011)	2. baseline		1A powerful combination outperforms the best single feature.. 2 Combining kernels outperforms the best single kernel. 3. Combinations of numerous levels yield superior results compared to single levels.
CLASSIFICATION IMPROVEMENT BASED ON FEATURE COMBINATION	Yeh,Lin,Chang(2012)	Discrete topic vector model.	1. The proposed feature processing flow is capable of achieving higher accuracy while maintaining the same ROC benchmark.	1. An key future topic is how to assign a unique cluster number to each feature. 2. The selection of an optimal degree parameter significantly impacts the time and space complexity.
SCENE CLASSIFICATION USING MULTI RESOLUTION LOW LEVEL FEATURES COMBINATION.	Li Zhou Zongtan Zhou,Dewen Hu(2013)	Uses GBPWHGO(descriptor)	1. This method demonstrates comparable performance to prior methods.	1. There is no requirement to divide the scene image into segments or convert local descriptors into a codebook. 2. The descriptor exhibits a high level of discriminative ability.
A SIMPLE FEATURE COMBINATION	Jian Hou, Marcello Pelillo(2013)	Based on dominant set clustering	1. It outperforms the CV approach and is the most effective single	1. Analysing the link between the SVM classification mechanism and

---

TITLE	AUTHOR & YEAR	TECHNIQUES IMPLEMENTED/METHODS	FINDINGS/RESULT	FUTURE WORK/CONCLUSION
METHOD BASED ON DOMINANT SETS.			classifier compared to MKL. 2. Doesn't always utilize training kernel matrix as whole.	the clustering method of dominating sets. 2a novel method to evaluate the discriminative power of kernels.
IMPROVE SCENE CLASSIFICATION BY USING FEATURE & KERNEL COMBINATION	Lin Yuan, Li Zhou, Dewen Hu(2014)	GBPWHGO+LBP+SIFT descriptor	1. The combined feature incorporates additional information to maintain invariance in terms of illumination, rotation, and scale. 2. The suggested technique combines different resolutions to create the ultimate classifier.	1. Integrates three widely used local descriptors that encompass a greater amount of local information. 2. Increased capacity to handle variations in scale.
FEATURE LEVEL COMBINATION FOR OBJECT RECOGNITION	Soofivand, Amirkhani, Daliri, Rezaeirad(2014)	SIFT+LBP+PHOG+GIST		1. Improves performance on Caltech-101 data set. 2.can be applied to many image data sets
DISCRIMININITIVE FEATURE COMBINATION SELECTION FOR ENHANCING MULTICLASS CLASSIFICATION.	Aibo Song, Wei Qian, Zhiang Wu & Jinghua Zhao2015	MFP_FS algorithm using Naïve Bayes(NB) and C4.5	overall best performance	1. Uses both labelled and unlabeled cases to identify composite featurtes that have a higher level of discrimination.
HIGHER LEVEL FEATURE COMBINATION VIA MULTIPLE KERNEL LEARNING FOR IMAGE CLASSIFICATION	Wei Luo, Jian Yang, Wei Xu, Jun Li,Jian Zhang(2015)	SSaC	MKL-*l2* achieves better performance	1.SSaC alleviates the information suppression problem 2. Further studied about the robustness of MKL method.
FEATURE COMBINATION AND THE KNN FRAMEWORK IN OBJECT CLASSIFICATION	Hou, Gao, Xia & Qi(2016)	A new weighted average combination method	Can be seamlessly included into the K-nearest neighbours (KNN) architecture.	Dominant sets have a promising future as clustering can be extended to that.
FEATURE COMBINATION via CLUSTERING	Jian Hou & Huijun Goa Xuelong Li(2017)	Clustering(dominant set algorithm)	Produces substantial improvement in performance compared to the most advanced algorithms currently available.	1. Combines only kernels suitable for combination. 2. corresponds to sparse solution of MKL and LP-β

TITLE	AUTHOR & YEAR	TECHNIQUES IMPLEMENTED/METHODS	FINDINGS/RESULT	FUTURE WORK/CONCLUSION
				3. The algorithm can be applied where memory space is not large.

4. A Graphical Comparative Study

Having gone through different papers, inspecting the pros and cons of different methods in detail, here we provide a graphical comparison among three chosen principal methods. They are average combination method, MKL and Dsets clustering. The reason behind choosing these three methods is twofold. First is that the average and MKL have been included almost in every paper to carry out experiments thus referred as base methods. Second is these three methods are promising. Although Dset is a recent and lacks large-scale works, but because of its eye seeking performance and for being a future oriented method, we have added this. The graph compares the three methods in terms of recognition rates as obtained from the two relevant papers [24] & [26] over a common dataset i.e. Flower-17 [28].

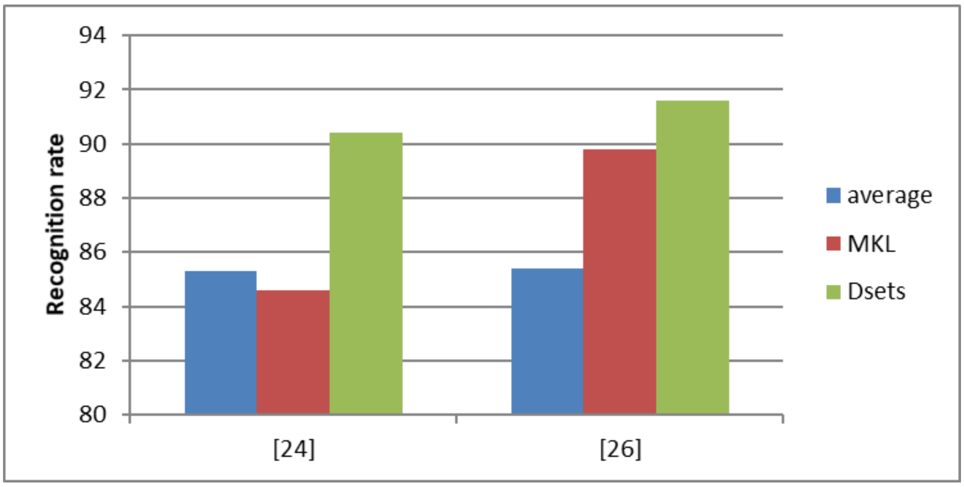

Fig. 6 Comparison of average, MKL and Dsets

The graph signifies that the baseline method produces uniform performance as compared to the sophisticated MKL method. The Dsets clustering has high performance than the other two methods which implies that a lot of future work could be possible on this.

5. Conclusion

Our paper presents a systematic literature survey on feature combination, its methods, advantages and limitations related to every individual method and finally a graphical comparison among the dominant works. Though innumerable algorithms have been executed successfully as per various papers, it's difficult to set a benchmark for sheer examination of these methods. For the time being, algorithms providing high performance as well as low computation load and consuming low memory space are considered for further research and alterations

Refrences

[1] Xu Cheng Yin, Chang –Ping Liu and Zhi Han, "Feature combination using boosting" X.-C. Yin et al. / Pattern Recognition Letters 26 (2005) 2195–2205

[2] P. Gehler and S. Nowozin, "On feature combination for multiclass object classification," in *Proc. IEEE 12th Conf. Comput. Vis.*, Kyoto, Japan, Sep./Oct. 2009, pp. 221–228.

[3] Linbo Zhang, Baihua Xiao, Chuncheng Wang, Gang Cheng and Yunxue Shao, "An efficient strategy for features combination" 2010 3rd International Congress on Image and Signal Processing (CISP2010)

[4] J. Hou, B.-P. Zhang, N.-M. Qi, and Y. Yang, “Evaluating feature combination in object classification,” in *Proc. Int. Symp. Vis. Comput.*, Las Vega, NV, USA, Sep. 2011, pp. 597–606.

[5] Jian Hou, Wei-Xue Liu and Hamid Reza Karimi, “Exploring the best classification from average feature combination” Volume 2014, Article ID 602763, 7 pages

[6] G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. El Ghaoui, and M. I. Jordan, “Learning the kernel matrix with semidefinite programming,” *J. Mach. Learn. Res.*, vol. 5, pp. 27–72, Jan. 2004.

[7] A. Kumar and C. Sminchisescu, “Support kernel machines for objectrecognition,” in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Rio de Janeiro,Brazil, Oct. 2007, pp. 1–8.

[8] Y.-Y. Lin, T.-L. Liu, and C.-S. Fuh, “Local ensemble kernel learning forobject category recognition,” in *Proc. IEEE Conf. Comput. Vis. PatternRecognit.*, Minneapolis, MN, USA, Jun. 2007, pp. 1–8

[9] M. Varma and D. Ray, “Learning the discriminative power-invariance trade-off,” in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Rio de Janeiro,Brazil, Oct. 2007, pp. 1–8.

[10] J. Yang, Y. Li, Y. Tian, L. Duan, and W. Gao, “Group-sensitive multiple kernel learning for object categorization,” in *Proc. IEEE 12th Int. Conf.Comput. Vis.*, Kyoto, Japan, Sep./Oct. 2009, pp. 436–443.

[11] C.Cortes, M.Mohri, A.Rostamizadeh, “Learning non-linear combinations of kernels”, in: Advances in Neural Information Processing Systems, 2009, pp.396–404.

[12] M.Varma, and B.R.Babu, “More generality in efficient multiple kernel learning”, in: International Conference on Machine Learning , 2009, pp.1065–1072.

[13] S.V.N.Vishwanathan, Z.Sun, N.T.Ampornpunt, M.Varma, “Multiple kernel learning and the SMO algorithm”,in: Advances in Neural Information Processing Systems, 2010, pp.2361–2369.

[14] F.R.Bach, “Exploring large feature spaces with hierarchical multiple kernel learning”, in: Advances in Neural Information Processing Systems, 2008, pp. 105–112.

[15] Wei Luo, Jian Yang, Wei Xu, Jun Li and Jian Zhang, “Higher –level feature combination via multiple kernel learning for image classification”

[16] Jian-hua Yeh, Chen Lin, Yuan-ling Chang, ‘Classification improvement based on feature combination and topic vector model.” 2012 International Conference on Systems and Informatics (ICSAI 2012)

[17] Li Zhou, Zongtan Zhou and Dewen Hu, “Scene classification using multi resolution low level feature combination” L. Zhouetal./Neurocomputing122(2013)284–297

[18] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, New York, NY, USA, Jun. 2006, pp. 2169–2178.

[19] A. Oliva and A. Torralba, “Modeling the shape of the scene: A holistic representation of the spatial envelope,” *Int. J. Comput. Vis.*, vol. 42, no. 3, pp. 145–175, 2001.

[20] J. Wu and J. M. Rehg, “Beyond the Euclidean distance: Creating effective visual codebooks using the histogram intersection kernel,” in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Kyoto, Japan, Sep./Oct. 2009, pp. 630–637

[21] J. Hou and M. Pelillo, “A simple feature combination method based on dominant sets,” *Pattern Recognit.*, vol. 46, no. 11, pp. 3129–3139,2013.

[22] M. Pavan and M. Pelillo, “Dominant sets and pairwise clustering,” *IEEETrans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 167–172, Jan. 2007.

[23] Jian Hou, Huijun Gao, Qi Xia and Naiming Qi, “Feature combination and the kNN Framework in object classification” ieee transactions on neural networks and learning systems, Vol.27, No.6,June 2016

[24] Jian Hou, Huijun Gao and Xuelong li, “Feature combination via clustering” ” ieee transactions on neural networks and learning systems, 2017.

[25] S. R. Bulò, M. Pelillo, and I. M. Bomze, “Graph-based quadratic optimization: A fast evolutionary approach,” *Comput. Vis. Image Understand.*, vol. 115, no. 7, pp. 984–995, 2011.

[26] Lin Yuan, Fanglin Chen, Li Zhou and Dewen Hu, “Improve scene classification by using feature and kernel combination.

[27] Mehrdad Ahmadi Soofivand, Abdollah Amirkhani, Mohammad Reza Daliri, Gholamali Rezaeirad "Feature level combination for object classification" 2014 4th conference on International Conference on Computer and Knowledge Engineering.
[28] Taiwan Huang, Lei Li, Yazhao Zhang, Junqi Chi "Summarization based on multiple feature combination"
[29] Aibo Song, Wei Qian, Zhiang Wu, Jinghua Zhao "Discriminative Feature combination Selection For Enhancing Multiclass Classification." 2015 International Conference on Behavioral, Economic and Socio-Cultural Computing.
[30] D. G. Lowe, "Distinctive image features from scale-invariant keypoints,"*Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[31] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, San Diego, CA, USA, Jun. 2005, pp. 886–893.
[32] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," *Comput. Vis. Image Understand.*, vol. 110, no. 3, pp. 346–359, 2008
[33] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002
[34] L.-J. Li and L. Fei-Fei, "What, where and who? Classifying events by scene and object recognition," in *Proc. IEEE 11th Int. Conf. Comput.Vis.*, Rio de Janeiro, Brazil, Oct. 2007, pp. 1–8.
[35] Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories.In: CVPR, Workshop on Generative-Model Based Vision, p. 178 (2004)
[36] O. M. Parkhi, A. Vedaldi, A. Zisserman, C. V. Jawahar, "cats and dogs" IEEE conference on computer vision and pattern recognition, 2012.