Research Article

SOHAG JOURNAL OF SCIENCES

https://doi.org/10.21608/sjsci.2023.212137.1081

Content-Based Image Retrieval Using BRISK and SURF as Bag-of-Visual-Words for Naïve Bayes Classifier

Samy Bakheet*, Mahmoud Mofaddel, Emadeldeen Soliman and Mohamed Heshmat

Faculty of Computers and Artificial Intelligence, Sohag University, Sohag 82524, Egypt.
**E-mail: samy.bakheet@fci.sohag.edu.eg*

Received: 20th May 2023, **Accepted:** 7th June 2023
Published online: 8th June 2023

Abstract: For Content-Based Image Retrieval (CBIR) systems, there are numerous ways. However, the results from the single feature kind are not sufficient. In this paper, The Adaptive Feature Fusion for the Naïve Bayes classifier (AFF-NB) framework is proposed. The local features are constructed from the fuse of the Binary Robust Invariant Scalable (BRISK) and the Speeded-Up Robust Features (SURF) detectors. The local features are then adaptively fused. The Gaussian Mixture Models (GMM) clustering algorithm clusters the extracted features to build a visual words codebook. Then a feature, quantization algorithm via the Cosine Distance Matrix (CDM) is applied to construct the Bag-of-Visual-Words (BoVW). To minimize the risk of data overfitting, the BoVW is normalized. The retrieved images are sorted according to how closely they resemble the query image using an inverted index strategy based on the CDM. The results demonstrate that the suggested technique increases CBIR accuracy to 96.8% on the commonly used Caltech-10 dataset.
Keywords: CBIR, SURF, BRISK, BoVW, Naïve Byes Classifier.

1. Introduction

CBIR stands for image retrieval based on the noticeable things in the images. Due to the increase in the storage capacities of image databases and the accessibility of digital image-capturing devices, it has recently become a focus of active research. CBIR approaches are becoming effective and efficient for managing and retrieving images from massive image collections in comparison to conventional text-based image retrieval (TBIR) techniques. The CBIR-based approaches took the place of the conventional TBIR-based approaches, which had several drawbacks, including a poor representation of the image's salient objects, a reliance on a particular language, a high computational cost, and a lengthy manual annotation process in dealing with large image databases [1].

The CBIR is used in many different contexts, including medical fields, criminal prevention, book retrieval from digital libraries, and fingerprint identification and retrieval. In CBIR, features are used to describe the objects in the images. To find the most similar images, it measures and extracts information from both the database's images and the query image. In CBIR, low-level characteristics are typically utilized to identify image objects in their active state. Texture, shape, and color features are the most often utilized low-level features in CBIR. The texture-based characteristics consider variations in pixel intensities to accurately depict the image's salient items as perceived by humans [2].

The BoVW model is a popular technique used in content-based image retrieval (CBIR) because it captures the local features of an image in a way that is robust to changes in scale, rotation, and deformation. In contrast, color-based characteristics, such as color histograms and color moments, are sensitive to variations in lighting conditions, camera settings, and color distribution in the image. So, the BoVW models are now receiving big attention in CBIR.[3]

The BoVW model represents an image as a collection of local features, which are extracted using techniques such as SIFT, SURF, or ORB. These features are then clustered into a codebook, and the image is represented as a histogram of the codebook entries. This histogram captures the spatial distribution of the local features, which can be used to match the image with other images in a CBIR system.

For flawless retrieval results in CBIR, image retrieval based on a single feature has not yet been shown to be useful. Therefore, various image features are combined to improve the effectiveness of image retrieval. Local features are more reliable and work better at recognizing objects even under extreme clutter and occlusion because they visualize the image in localized patches [4].

In this paper, to obtain the complementary features that improve CBIR performance, BRISK and SURF feature detection algorithms are used to extract and describe the image's local features. The suitable feature vectors are then picked as samples from all images using the FeatureWiz selection technique. The GMM clustering technique is used to form the visual words codebook. Vector quantization is then used to define each image in this codebook as a one-dimensional feature vector. The vector quantization produces a distinct feature vector for each image in the dataset. The robustness of the visual words in the suggested method outperforms recent CBIR methods.

These are the following sections of this paper. The literature review of CBIR techniques is described in Section 2. The

suggested strategy is described in Section 3. The experimental findings, running costs, and performance evaluation parameters of the suggested technique are all covered in Section 4. The concluding analysis of the suggested approach and future directions are discussed in Section 5.

2. Literature Review

Over the past thirty years, content-based image retrieval (CBIR) has emerged as a vibrant area of research. In recent times, numerous methods have been introduced in the field of computer vision and image classification applications that utilize multi-feature fusion [5]. This section presents a cross-sectional analysis of related work by comparing the different methods employed in previous studies for each step of the CBIR system.

Using color and texture data, Kavitha et al. [6] suggested a CBIR technique. Hue Saturation Value (HSV) and image color moments are employed as the characteristics of the image. Gabor texture descriptors are used as the texture's features. Users give weights to each feature in turn and compute the degree of similarity with the combined features of texture and color using normalized Euclidean distance.

Bakheet et al. [7] proposed a CBIR based on keypoint feature fusion and hybrid of BoVW. They utilized FeatureWiz Selection for feature selection process. Their work involves a comprehensive evaluation of the nine most important keypoint feature detection in conjunction with seven robust feature descriptors algorithms. The SURF and FAST keypoints are fused, then described by RootSIFT. FeatureWiz Selection is applied to choose the best features. K-Means algorithm clusters those features to visual words for the SVM learning model.

Lux et al. [8] unveiled the open source, lightweight LIR (Lucene Image Retrieval) Java library for CBIR. It includes methods for indexing and retrieval as well as common and modern global image attributes. It can be readily integrated into applications without the need for a database server because it is built on a lightweight embedded text search engine.

Jabeen et al. [9] proposed a CBIR method based on the visual word fusion of fast retina keypoint (FREAK) and sped-up robust features (SURF) feature descriptors. FREAK is a dense description whereas SURF is a sparse one. Additionally, the scale and rotation-invariant descriptor SURF outperforms other descriptors in terms of robustness, repeatability, and distinctiveness. It is resistant to noise, detecting mistakes, geometrical distortions, and photometric faults. Additionally, it outperforms the FREAK descriptor in images with low illumination. In contrast, FREAK, a fast descriptor inspired by the retina, outperforms the SURF descriptor in classification-based challenges.

Zhou et al. [10] presented CBIR approach involves combining three distinct features and optimizing the feature metric through a diffusion process. In order to improve the ability to distinguish between images, the chosen features are the local directional pattern, color histogram, and dense SIFT features based on bag-of-features (BoF). The diffusion process is utilized to achieve an image matching optimization based on hybrid features. This process can capture the underlying manifold structure within a dataset, leading to a significant improvement in overall retrieval performance. Additionally, a new search strategy is developed to further enhance the effectiveness of the diffusion process when working with a small number of retrieval images.

Khan et al. [11] proposed a method for improving image classification by creating a feature fusion vector that combines local tetra angle patterns and color features. In order to optimize the support vector machines (SVMs) used in the classification process, they employed Genetic Algorithms (GA). To measure the similarity between query images and those in the dataset, they utilized the Chi-square quadratic distance metric.

Alkhawlani et al. [12] introduced an image retrieval system that retrieves related images from common databases quickly and accurately using local feature descriptors and the BoVW model. The suggested approach creates image signatures that are rotation and scale-invariant by using SIFT and SURF techniques as local descriptors. Additionally, it builds a visual word for the feature descriptors acquired via local descriptors techniques using K-Means clustering algorithm. The SVM technique is used to quickly retrieve additional images that are pertinent to the query.

In order to enhance a CBIR system that relies on local features, Karakasis et al. [13] incorporated affine moment invariants with bag-of-visual-words (BoVW) technique. Suharjito et al. [14], on the other hand, utilized Gaussian mixture models (GMM) to create a Visual Words Codebook (VWC). They also proposed a multiclass SVMs) approach to classify images, using chi-square, Hellinger's, and linear kernel functions as similarity metrics. The similarity distance was computed based on the query image color histograms and images within the same class.

Hameed et al. [15] proposed a CBIR Based on Feature Fusion and SVM. where the combination of local and global features serves as a good image descriptor. Accuracy and retrieval time are the two main essential metrics used to assess CBIR performance. The first one has to do with how many images from the same semantic class were retrieved, whilst the second one has to do with how quickly the search was conducted. The image features description is based on the texture and color features. After converting the RGB channels to the moments' domain to create the color descriptor, the standard deviation, mean, and statistical moments are computed. Edge and texture descriptors are built using Canny Edge Detection and LBP.

Bakheet et al. [16] proposed a novel paradigm for sleepiness detection in drivers, in which an adaptive descriptor with the properties of uniqueness, compactness, and robustness is constructed using an enhanced version of HOG features based on shifted orientations binarized histograms. The trained Nave Bayes (NB) classifier receives the final HOG descriptor produced from the binarized HOG features to determine the driver's level of tiredness.

3. Methodology

The BoVW model serves as the foundation for the proposed method's methodology. In the BoVW model, features are first collected from the images; next, using a clustering technique, comparable portions of the images are grouped together to create a dictionary. Each focal point of a cluster serves as a visual word inside the dictionary. The histogram is created using the visual terms from each image in the dictionary. Histograms from the database's training images are used to train the classifier, while histograms from the database's test images are used to evaluate the effectiveness of the CBIR approach.

By comparing the histograms of the query image and the database images, the similarity between them is used to obtain the retrieved images. (Figure 1) displays the block diagram of the suggested approach, which is based on the feature detection fusion of BRISK and adaptive SURF to form BoVW for the Naïve Bayes classifier.

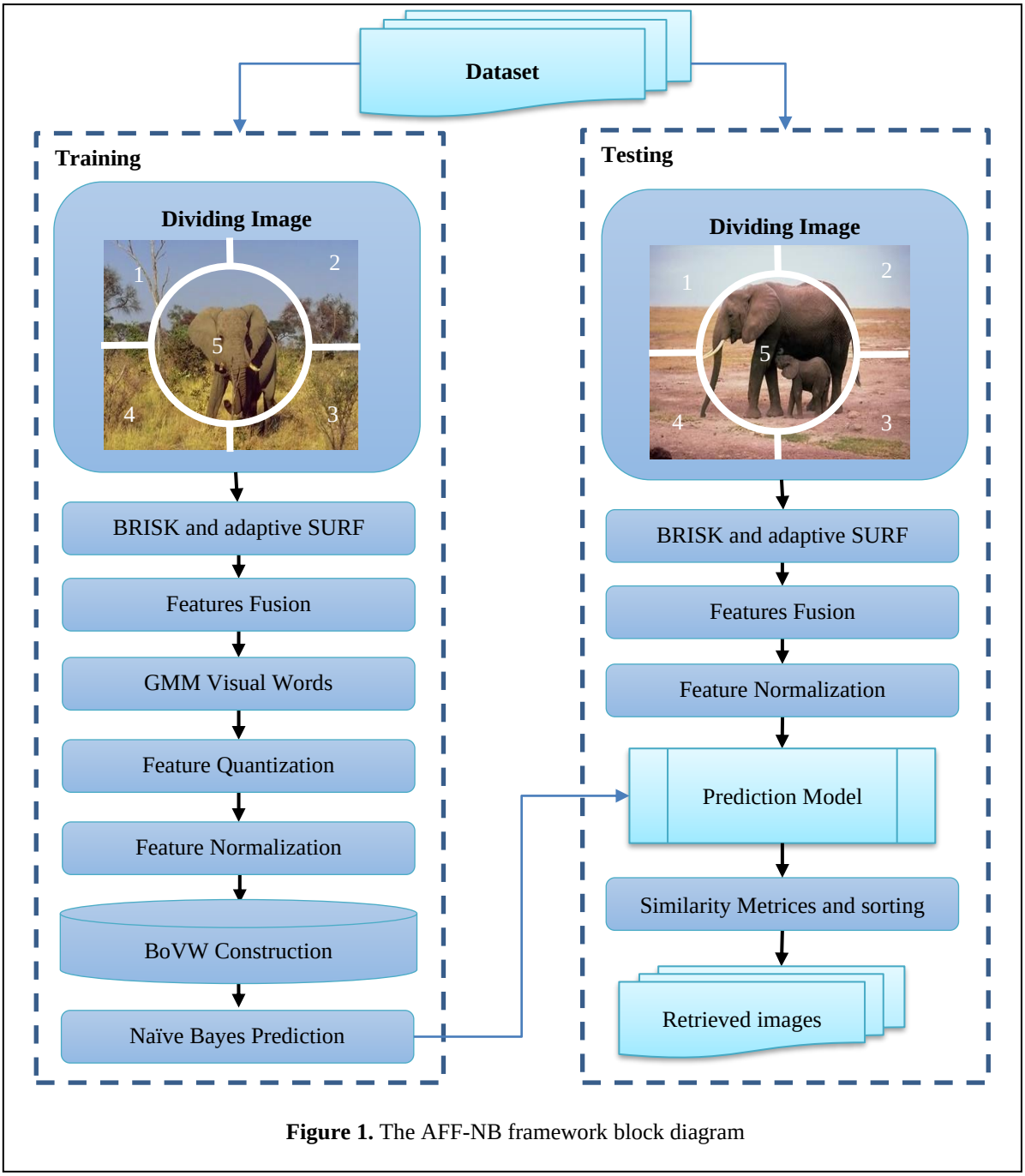

Figure 1. The AFF-NB framework block diagram

2.1. Feature Extraction

The main difficulty with the most of image classification approach is that each pixel in an image has its characteristics as color-value and unique pixel location. In this paper, to get over this problem, features for BRISK and enhanced SURF local features recognition are taken from five distinct areas of the image and combined with each other. The extracted features were then used to build a bunch of adaptive visual word using the FeatureWiz selection features and GMM clustering

algorithms. Fig. 3 illustrates an image split into five areas of interest.

2.1.1. Speeded-Up Robust Features

In contrast to BRISK, the SURF's feature point identification is a Hessian matrix based. SURF detects the keypoint feature and greatly speeds up computation by locating the Hessian matrix local maximum determinant and using the integral image [17]. The point X=(x,y) in the image's Hessian matrix is defined as follows on the scale σ:

$$H(x,\sigma) = \begin{pmatrix} L_{xx}(x,\sigma) & L_{xy}(x,\sigma) \\ L_{xy}(x,\sigma) & L_{yy}(x,\sigma) \end{pmatrix} \quad (1)$$

here $L_{xx}(x,\sigma)$ is the convolution of the Gaussian second order differential and the image I(x, y) at point $X = (x,y)$. $L_{xx}(x,\sigma)$ and $L_{yy}(x,\sigma)$ is similar to $L_{xx}(x,\sigma)$.

In order to increase computational speed, Bay et al. [18] recommended using a box filter to estimate the second-order partial derivatives of the Gaussian function, as well as an integral image to accelerate the convolution process. The aim of these techniques is to approximate the determinant of the Hessian matrix. [19]:

$$\det(H_{approx}) = D_{xx} \times D_{yy} - (0.9 \times D_{yy})^2 \quad (2)$$

where D_{xx}, D_{yy}, and D_{xy} are the approximate of L_{xx}, L_{yy}, and L_{xy}.

To construct a multiscale image pyramid, multiple box filter sizes can be utilized. SURF employs non-maximal suppression on each point in the multiscale image pyramid by comparing it with its 26 neighbors at both the current and neighboring scales. This process results in several candidate feature points. Interpolation is then performed to obtain the final feature points in both the scale space and the image space. To ensure rotational invariance, SURF computes the Haar wavelet characteristics in the feature point domain to determine the dominant orientation of each feature point.

2.1.2. Binary Robust Invariant Scalable Keypoint

BRISK utilizes a consistent sampling strategy to capture the region around the feature point. By creating concentric circles with varying radii around the feature point as the center, N sampling points are obtained through evenly spaced sampling on each circle. Gaussian filtering is then applied to the sample points of the concentric circles to avoid aliasing problems, as stated in [17]. As there are N sample points, these points are merged to form N(N-1)/2 point pairs, which are represented by a set A using Equation (3). The values of the smoothed intensity of the two sample locations after Gaussian filtering are $I_{(P_i,\sigma_i)}$ and $I_{(P_j,\sigma_j)}$, respectively. These values are utilized to calculate the local gradient $g(P_i,P_j)$ as Eq (4).

$$\mathcal{A} = \{(P_i,P_j) \in \mathbb{R}^2 \times \mathbb{R}^2 | i < N^j < i^i, j \in \mathbb{N}\} \quad (3)$$

$$g(P_i,P_j) = (P_i - P_j).\frac{I_{(P_j,\sigma_j)} - I_{(P_i,\sigma_i)}}{\|P_i - P_j\|^2} \quad (4)$$

here are two subsets of long-distance pairings $\mathcal{L}$ and short-distance pairings S, respectively:

$$\mathcal{L} = \{(P_i,P_j) \in \mathcal{A} | \|P_i - P_j\| > \delta_{min}\} \subseteq \mathcal{A} \quad (5)$$

$$S = \{(P_i,P_j) \in \mathcal{A} | \|P_i - P_j\| < \delta_{max}\} \subseteq \mathcal{A} \quad (6)$$

The above formula is utilized to estimate the primary direction of the feature point as being:

$$g = \begin{pmatrix} g_x \\ g_y \end{pmatrix} = \frac{1}{L}.\sum_{(P_i,P_j)\in\mathcal{L}} g(P_i,P_j) \quad (7)$$

where L is the number of $\mathcal{L}$, g_x is the gradient in the x-direction, and g_y is the gradient in the y-direction.

BRISK rotated the sampling region at $\alpha = \arctan(g)$ to keep rotation invariance. After rotation, the intensity values of the point pair (P_i^α, P_j^α) are compared in the new subset of short-distance pairings S to create the 512-bit bit-vector description. Each bit b is equivalent to:

$$b = \begin{cases} 1, & I(P_j^\alpha,\sigma_j) > I(P_i^\alpha,\sigma_i) \\ 0, & Otherwise \end{cases} \forall (P_i^\alpha, P_j^\alpha) \in S \quad (8)$$

2.1.3. Hybrid Image Features Description

To generate a Hybrid Image Features Description (HIFD), the output features of each BRISK and SURF were concatenated, selecting only those features that corresponded to the same keypoint. It should be noted that neither BRISK nor SURF alone is capable of providing a better feature descriptor for images with similar color, texture, and objects as compared to HIFD. Each image in the dataset is described by a single HIFD, which captures the salient regions of interest in the image.

2.1.4. Gaussian Mixture Model (GMM)

K-Means is arguably the most popular and widely used clustering algorithm. But there are certain drawbacks. One of them is that the clusters will be separated using an ideal radius value, determined using the Euclidean distance to the point, from the cluster center. Therefore, if the cluster is not described as having a circular form, it may be difficult to appropriately separate it. While the elliptical shape is the most popular, GMM also supports various shapes.

The GMM operate under the assumption that a specific number of Gaussian distributions exist, with each distribution representing a cluster. Consequently, GMMs tend to group data points that correspond to the same distribution.

2.2. Feature Quantization

In order to quantize the feature vectors of an image into a single BoVW histogram vector, Construct the CDM between the visual words vector and all of the extracted features for that image. The following outlines the process for computing the cosine distance between vectors A and B.

$$D_{(A,B)} = Cos(A,B) = \frac{A.B}{\|A\|\|B\|} \quad (9)$$

2.3. Visual Words Features Normalization

By shifting and rescaling values, the normalization scaling approach makes them lie between the ranges of 0 and 1. They

are sometimes referred to as Min-Max scaling. The normalizing stage lowers the likelihood of overfitting. Normalizing procedure is calculated as follow:

$$N = \frac{H - h_{min}}{h_{max} - h_{min}} \quad (10)$$

where N is the normalized features, while h_{max} and h_{min} are the maximum and minimum features for the Histogram H of the visual words features.

2.4. Naïve Bayes Classifier

The naive Bayes classifier is a probabilistic approach to classification that is based on the Bayes theorem and the assumption of feature independence. The classifier combines a naive Bayes probability model with a decision rule, as illustrated below.

$$\beta = argmax_{\omega_j \in c} P(\omega_j) \prod_i P\left(\frac{\alpha_i}{\omega_j}\right) \quad (11)$$

Here α_i is the i_{th} attribute and ω_j is the j_{th} class. $P(\omega_j)$ is the prior probability and $P(\alpha_i/\omega_j)$ is the posteriori probability. $P(\alpha_i/\omega_j)$ is the probability that attribute α_i occurs in an image given the image belong to class ω_j and c is the set of targets. The winning neuron's output and the total of all neurons' outputs are here two properties, while correct classification and incorrect classification are two classifications. The decision rule in Eq (11) can be used to decide if a categorization is valid or incorrect.

The Naive Bayes classifier aids in the dynamic selection of the search space size to be used during retrieval. The search space is limited to a single output class for each accurate classification predicted by Naive Bayes.

4. Results and Discussion

This section discusses the proposed AFF-NB framework and provides the experiment's results. On the Caltech-10 datasets, the effectiveness and accuracy are assessed. The effectiveness of the proposed AFF-NB was compared to the effectiveness of the BRISK and Improved-SURF feature descriptors based on CBIR. To obtain the average performance, each experiment was run five times using K-fold Cross-Validation (KCV). The suggested AFF-NB's performance was also evaluated in comparison to current state-of-the-art techniques.

4.1. Experimental Settings

The features parameter, performance and running time of any CBIR system are primarily assessed using the CBIR settings. Below is a description of the parameters for the various experiments for the suggested framework.

4.1.1. Features Parameter

The vocabulary was created using the characteristics that were retrieved. It is not advised to use all the retrieved characteristics as the clustering step's input. According to Datta et al. [20], although using more features improves outcomes, doing so raises processing costs and can lead to overfitting. Each vocabulary size that was employed was examined using different feature percentages (i.e., 10%, 25%, …,100%) of the total extracted unique features to determine which feature percentage provided the highest performance at the lowest processing cost. Based on FeatureWiz decision, the features were chosen.

4.1.2. Codebook length

The number of feature clusters represents the codebook length. One of the main criteria for assessing CBIR system is the codebook length. The system performs better with a larger codebook length, however long codebook tends to overfit and raise processing costs, and vice versa. To choose the best codebook length from the training features for effective vector quantization, a range of codebook lengths (i.e., 256, 512, 1024..., 32,780) were considered.

4.1.3. Data Splitting

Two groups of experimental datasets are created; one is for training and the other is for testing. Five-fold cross-validation is used to choose the training group. The dataset is split into two parts: 20% for testing and 80% for training.

4.2. Evaluation Concepts

The K-fold Cross-Validation (KCV) is a straightforward method for deciding which model to use and how accurate an estimator is. KCV serves as the main metric for assessing the effectiveness of the recommended CBIR architecture. The ideal range for k, according to [21] is between 5 and 10. The framework's effectiveness is assessed using 5-fold cross-validation. Each dataset class measures and summaries precision. The average precision across all classes is used to compute the CBIR system's overall precision. As long as the value of k is set to 5, 80% of the data will always be used to create the training model and 20% for model validation. Five validation attempts were made, with the average being calculated after each attempt (each time was for a different fold). Accuracy, precision, Mean Average Precision (MAP), recall, and F1-score are the assessment metrics that are employed.

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \times 100\% \quad (12)$$

$$Precision = \frac{TP}{TP + FP} \times 100\% \quad (13)$$

$$MAP = \frac{\sum_{q=1}^{Q} Ave(q)}{\mathrm{Q}} \times 100\% \quad (14)$$

$$Recall = \frac{TP}{TP+FP} \times 100\% \quad (15)$$

$$F1 - Score = 2\frac{Precision \times Recall}{Precision + Recall} \times 100\% \quad (16)$$

4.3. Datasets and Results

The Caltech-10 dataset, containing 1000 images belonging to 10 semantic classes, each with 100 images, was utilized to evaluate and test the AFF-NP framework. This dataset is a subset of the Caltech-101 dataset, which comprises 101 classes. We formed the Caltech-10 dataset by selecting specific classes, including Faces, , Bonsai Leopards, Ketch, Motorcycles, Brain, Car side, Watch, Airplanes, and Grand piano, from Caltech-101. To

assess the impact of two parameters, BFSP and codebook size, on the performance of the proposed framework, we tested a combination of BoVW-Feature Selection Percentage (BFSP) and codebook length, as presented in Table 1. (Figure 2) shows Caltech-10 datasets as a sample image of each semantic class.

The 75% for BFSP and 2048 codebook length combination had the highest accuracy rating (96.80%). (Figure 3) displays the BFSP and codebook length combination on the Caltech-10 dataset. Table 2 results demonstrate that the suggested AFF-NB framework, built on HIFD features and uses BFSP, performs better than independent feature descriptors like BRISK and SURF.

Table 1. The AFF-NB Performance on different codebook lengths and BFSP on the Caltech-10 dataset

BFSP %	MAP values in percent on various codebook lengths								
	256	512	1024	2048	4096	8192	16,384	32,768	65,536
10%	82.96	89.01	92.77	92.29	94.32	95.59	93.16	92.25	92.00
25%	93.27	94.28	95.26	95.43	96.20	96.44	95.57	94.62	93.89
50%	93.34	94.98	96.10	96.10	96.48	96.46	96.12	95.42	94.11
75%	95.08	96.47	96.72	**96.80**	96.34	96.37	96.20	95.02	94.80
100%	93.01	94.07	93.98	94.83	94.22	94.46	94.4	93.33	92.76
MAP	91.53	93.76	94.97	95.10	95.51	95.86	95.09	94.13	93.51
SE	2.174	1.260	0.716	0.779	0.509	0.387	0.580	0.586	0.501
SD	4.862	2.818	1.600	1.742	1.139	0.864	1.297	1.311	1.119

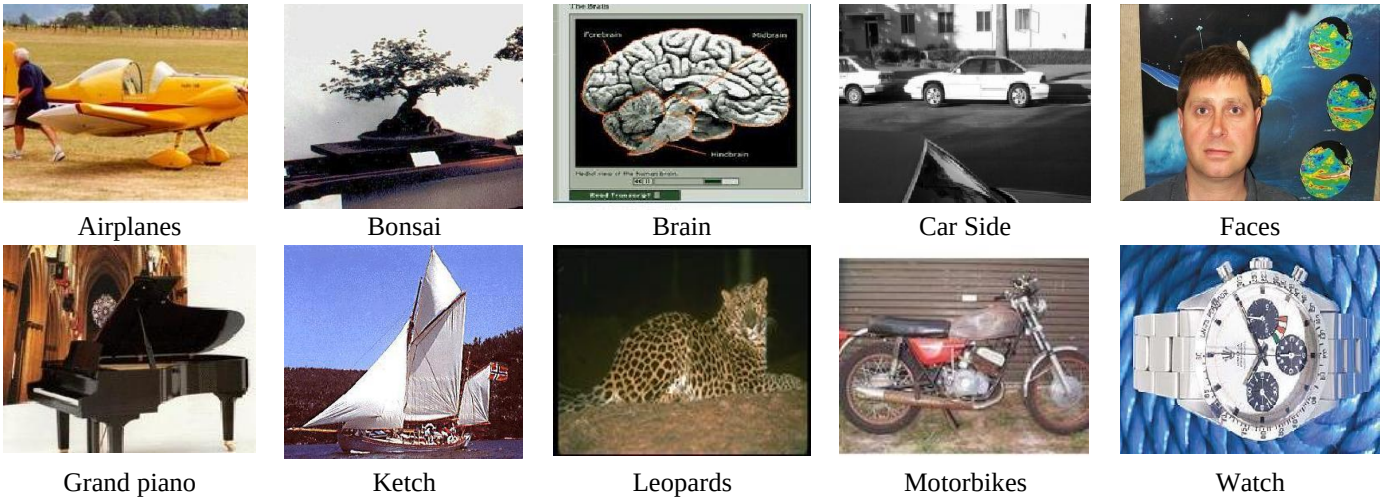

Figure 2. Caltech-10 dataset sample images

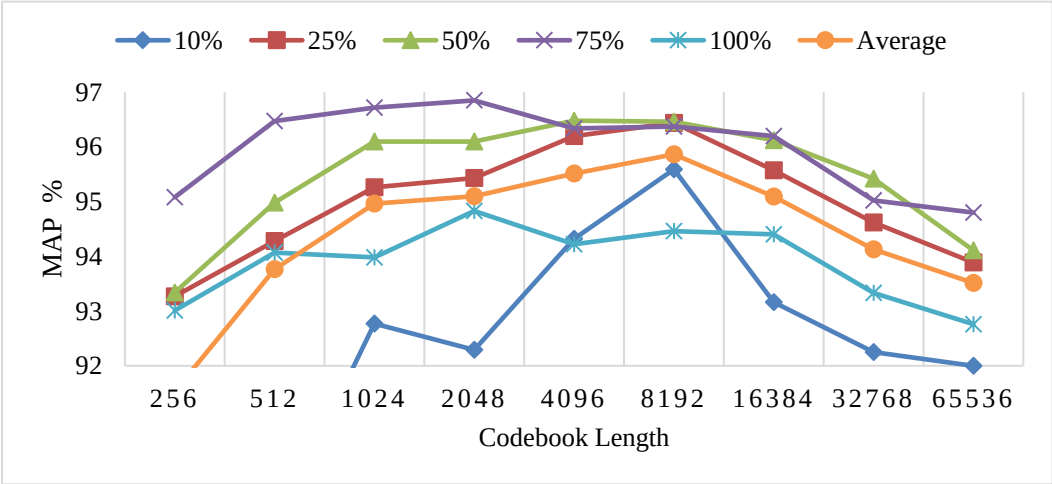

Figure 3: AFF-NB on various codebook lengths and BFSP on the Caltech -10 dataset

Table 2. AFF-NB vs. standalone SURF and BRISK on Caltech-10 dataset classes

Classes	SURF			BRISK			Proposed AFF-NB		
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
Watch	96	18	30	95	19	32	98	19	32
Leopards	95	18	30	95	18	30	98	19	32
Car side	88	18	30	90	17	29	95	19	32
Airplanes	90	18	30	87	17	29	96	18	30
Bonsai	87	16	27	85	16	27	97	19	32
Grand piano	90	17	29	89	18	30	98	19	32
Faces	90	18	30	87	17	28	97	18	30
Motorbikes	89	17	29	90	18	30	95	18	30
Brain	88	17	28	86	17	29	96	18	30
Ketch	90	18	30	89	18	30	98	19	32
Average	90.30	17.50	29.31	89.30	17.52	29.29	**96.80**	18.64	31.26

5. Conclusions

In this paper, an adaptive AFF-NB framework of the Content-Based Image Classification and Retrieval (CBICR) based on the Naïve Bayes classifier was presented. In this framework, a set of features were conveniently detected and adaptively fused. The experimental results obtained on the Caltech-10 benchmark dataset unveiled that the fusion between the BRISK and SURF features did not only positively impact the performance of the proposed AFF-NB approach for CBICR, but also narrowed the semantic gap between human-level semantic vision concepts and low-level automatic algorithms features. As a future work, we look forward to improving the performance using the Artificial Bee Colony (ABC) algorithm and evaluating the method on a larger scale dataset to provide a more detailed analysis for general frameworks with a web interface.

5. References

[1] S. Bakheet, M. Mofaddel, E. Soliman, M. Heshmat, *Applied Mathematics & Information Sciences*, 14 (2020) 699–708.

[2] S. Sadek, A. Al-Hamadi, B. Michaelis, U. Sayed, *World Academy of Science, Engineering and Technology*, 57 (2009) 139–142.

[3] A. Jadia, M.P.S. Chawla, *International Journal of Innovative Science and Modern Engineering (IJISME)*, 6 (2020) 7-13.

[4] M.A. Shukran, M.N. Abdullah, M.S. Yunus, *Journal of Materials Science and Chemical Engineering*, 9 (2021) 51–57.

[5] S. Bakheet, M. Mofaddel, E. Soliman, M. Heshmat, *International Journal Of Engineering Research & Technology (Ijert.*, 9 (2020) 499–505.

[6] C. Kavitha, B.P. Rao, A. Govardhan, *International Journal of Computer Applications*., 2011, 15(7), 33–37.

[7] S. Sadek, A. Al-Hamadi, E. Soliman, M. Heshmat, *Sensors*, 23 (2023) 1653–1677.

[8] M. Lux, S. A. Chatzichristofis, *Proceedings of the 16th ACM international conference on Multimedia*, (2008) 1085–1088.

[9] S. Jabeen, Z. Mehmood, T. Mahmood, T. Saba, A. Rehman, M.T. Mahmood, *PloS one*, 13 (2018) e0194526.

[10] J. Zhou, X. Liu, W. Liu, J. Gan, *Multimedia Tools and Applications*, 78 (2019) 6163–6190.

[11] G. Yu, Y. Wang, J. Wang, C. Domeniconi, M. Guo, X. Zhang, *Information Fusion*, 63 (2020) 153–165.

[11] U. A. Khan, A. Javed, *Journal of King Saud University-Computer and Information Sciences*, 34 (2022) 7856-7873.

[12] M. Alkhawlani, M. Elmogy, H. Elbakry, *Int J Adv Comput Sci Appl*, 6 (2015) 212–219.

[13] E.G. Karakasis, A. Amanatiadis, A. Gasteratos, S.A. Chatzichristofis, *Pattern Recognition Letters*, 55 (2015) 22–27.

[14] A. Suharjito, D.D. Santika, *ICIC Express Letters*, 11 (2017) 1479–1488.

[15] I.M. Hameed, S.H. Abdulhussain, B.M. Mahmmod, A. Hussain, *14th International Conference on Developments in eSystems Engineering (DeSE)*, (2021) 552–558.

[16] S. Bakheet, A. Al-Hamadi, *Brain Sciences*, 11 (2021) 240.

[17] T. Huang, Z. Yu, X. Lin, L. Jiang, D., Zhao, *IT Convergence and Security.*, Springer, Singapore, (2017) 147-155.

[18] H. Bay, T. Tuytelaars, L.V. Gool, *Comput. Vis. Image Underst*, 110 (2006), 404–417.

[19] S.H. Cheon, I.K. Eom, S.W. Ha, Y.H. Moon, *Journal of Real-Time Image Processing*, 16 (2019) 1177-1187.

[20] R. Datta, D. Joshi, J. Li, J.Z. Wang, *ACM Computing Surveys (Csur)*, 40 (2008) 1–60.

[21] T. Hastie, R. Tibshirani, J. Friedman, *Data Mining, Inference and Prediction*, Springer, (2009) 214–215.