Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 10, October-2020

A Survey on Content-based Visual Information Retrieval

Samy Bakheet
Computer Science
Faculty of Computers and Information (FCI)
Sohag, Egypt

Mahmoud Mofaddel
Computer Science
Faculty of Computers and Information (FCI)
Sohag, Egypt

Emadedeen Soliman
Computer Science
Faculty of Computers and Information (FCI)
Sohag, Egypt

Mohamed Heshmat
Computer Science
Faculty of Computers and Information (FCI)
Sohag, Egypt

***Abstract*— Images have always been seen as an effective medium for visual data presentation. In recent years, a tremendous combination of images and videos have been grown up rapidly due to technology evolution. Content-Based Visual Information Retrieval (CBVIR), which is the process of searching for images via the end user's predefined specific pattern (hand sketch, camera capture, or web scrawled). CBVIR is still far away from achieving objective satisfaction due to image content-based search engines (for ex. Google image-based search) still not completely satisfying. This problem occurs because of the semantic gap between low and high visual level features representation of the image. In this paper, The state-of-art CBVIR techniques for multi-purpose applications are survived. The architecture of the promising CBVIR pipelines in recent decades, which witness the arising of computer vision is highlighted. Mathematical, machine, and deep learning-based CBVIR systems are introduced. Although the high computational cost of deep learning techniques remains the most efficient to utilize.**

Keywords*— CBIR, BOVW, Color histogram, Machine learning.

I. INTRODUCTION

The users are familiar with text search engines for searching for a text, but what about if a user wants to search by image content? The searching process may use meta-keywords which describe the query image. Searching by text requires a Text-Based Image Retrieval (TBIR) system. TBIR is not satisfying because of the limitations, such as the inability to thoroughly explain the meaning of the image content [10]. It is impossible to search for none tagged image using a TBIR system. Therefore, Content-Based Visual Information Retrieval (CBVIR) system is preferred to remove these limitations. Text-Based domain provides results with semantic similarity. In contrast, content-based search returns results with visual similarity. CBVIR is a simplified approach to search for an image in a large dataset based on the image content. CBVIR also is known as Query By Image Content (QBIC) or Content-Based Image Retrieval (CBIR) and is the other face for image classification. Along the progress way of developing an efficient CBVIR system, researchers introduced a high contribute papers in this field [21, 28, 29]. In the same way, a collection of surveys published by CBVIR communities Ease of Use [15, 23, 24, 26].

A lot of adaptive algorithms are developed for retrieving images from a large dataset. TBIR methods often use traditional database algorithms to manage text-based systems. TBIR uses meta-keywords, tags, etc. to describe the contents and features of an image. On the one hand, TBIR system is fast, and the actual images itself are examined if they are well labeled. On the other hand, TBIR has some limitation due to The image content is much more precious than what any set of keywords can express, which is known as the semantic gap between the descriptor and the image content. The bad part for TBIR systems is its limitation where it is not able to search inside an unannotated image dataset. Sometimes the same word can have several meanings in different contexts [21]. A picture is worth a thousand words. The best example of TBIR system is Flickr system.

CBVIR has been one of the most vivid research areas in the field of computer vision over the last ten years. That allows searching for an image by example or depending on the image content textures, colors, shapes, or any other information derived from the image itself. The greatest challenge in this technique is the Semantic gap between low-level information extracted by a machine from the image and what the image means to someone as a high-level. Any system that uses CBVIR technique must have an image descriptor to extract features from each image to indexes the dataset. Features are the output of an image descriptor. When inputting an image to the descriptor, the features will get out on the other end. In the most basic terms, features (or feature vectors) are just a list of numbers used to represent and quantify the image abstractly. Image descriptors examine the three aspects of the image content, where the descriptor produces a feature or multi-feature vectors to quantify shapes, colors, or textures of an image, sometimes maybe a combination of the three aspects.

1. Texture

Texture quantifies and represents the fellness, appearance, or consistency of a surface. Texture is the preferred choice to compare between roughness and smoothness. Texture refers to visual patterns of homogeneity. So, describing the texture of an image may be used to distinguishes between rocks, sand, and bricks. Texture technique more used in Tamura representation. Figure 1 illustrates various types of textures for

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 10, October-2020

brick, fingerprint, clouds. Texture descriptor is most used in image classification for CBVIR systems [16]. Haralick texture feature algorithm is efficient for texture descriptors.

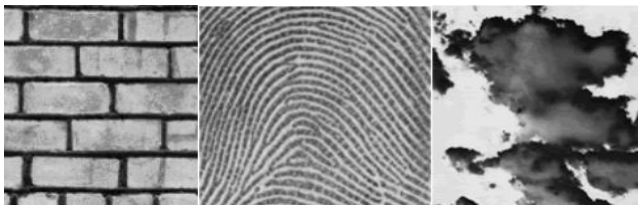

Figure 1. Texture paterns

2. color

Color space is just a specific organization of colors that allows us to represent and reproduce colors consistently. There were many color spaces, but the more related color spaces to CBVIR are (RGB, HSV, L*a*b, and grayscale, which is not technically a color space). Choosing a suitable color space plays a strict rule in CBVIR systems, and that depends on the used dataset. Color channels features histogram appears intensively during build CBVIR systems. Figure 2 shows RGB color histogram.

3. Shape

Shape refers to a particularly interesting region of an image. Shape descriptor extracts the contour of an object in an image. Contour is a curve joining all the consecutive points (along the boundary), which have the same color or intensity. Shapes could be outlined by applying segmentation or edge detection to an image. Shape descriptors should be invariant to translation, scale, and rotation. Some of the popular shape descriptors are Hu Moments, Zernike Moments, or Histogram of Oriented Gradients (HOG) algorisms.

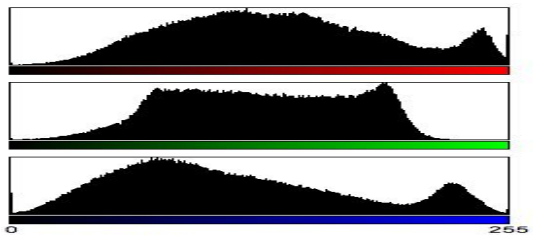

Figure 1. RGB color histogram

Region-Based Query In this aspect, the query may be an image itself or a region of an image. Then the system searches in the dataset for the more similar images of the query or the image that contains the image region query. In this case, the challenge is not to search for an object like a cat, chare, tree, etc. The challenge is to search for a region of interest in an image or the image itself.

Object-based Query In this aspect, Image retrieval systems retrieve images from a dataset based on the appearance of the physical objects in those images. The objects could be elephants, stop signs, helicopters, buildings, faces, or any other object that the user wishes to find [22]. This type of query is more related to image classification where machine learning is used, such as Support Vector Machines (SVMs) or Conventional Neural Networks (CNNs) algorithms to classify images depends on objects. Such image retrieval systems are generally successful for objects that can be easily separated from the background and that have distinctive colors or textures [22].

How are CBVIR and machine learning/image classification are different? Each of the branches studies images features descriptors. Each branch gives accuracy and relevancy evaluation notes. Machine learning includes techniques to make computers do intelligent human tasks such as recognition, classification, prediction, etc. Machine learning produces algorithms able to processes smart jobs without being explicitly programmed. From the other face, CBVIR uses machine learning methods for vector dimensionality optimizing and clustering, but these systems do not have actual learning. The gist of deference is that CBVIR does not understand or explain image contents but tries to quantify feature vectors extracted from images. Images with similar feature vectors are nearest to have related visual contents. CBVIR systems are capable of returning the nearest related images without knowing image contents, without require labeled data for learning. On the other hand, machine learning and image classifier need a set of labeled data to learn the system what each visual object in the dataset looks like. which may sometimes be a step of CBIR system. CBVIR systems may be considered as a dumb image classifier that has no notation of labels to make it more intelligent.

The reminder sections of this paper are structured as follows. Section 2 briefly illustrates the pipelines of a CBVIR system. Section 3 has a look at datasets and evaluation metrics, but in section 4 deeps into CBVIR various approaches. Finally, Section 5 concludes and discussions this paper.

II. ARCHITECTURE OF A CBVIR SYSTEM

There are many CBVIR methods; they all use the following four main phases, regardless of the used method. When designing the CBIR system, each of these phases is critical.

A. *Defining an image descriptor*

In this phase, the descriptor technique that will be used to extract features from an image have to be selected. It may use colors of the image; it may depend on the shape of an object or finds the characterize textures in the image. In other situations, descriptors use hybrid of color, shapes, and texture to describe features of an image. The descriptor uses a keypoint detector to determine the interesting points, then describes each of these points as a combination of its surrounding points. Figure 3 illustrate this process.

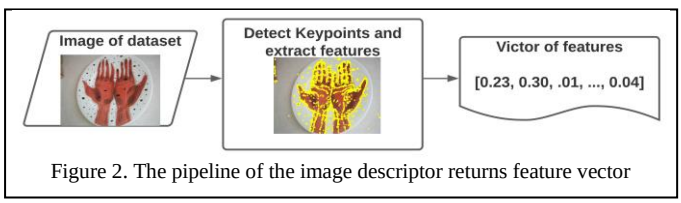

Figure 2. The pipeline of the image descriptor returns feature vector

B. *Indexing a dataset*

After the image descriptor extracts the features from each image in the dataset, the extracted features must be indexed in a suitable and fast access structure file or database. The

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 10, October-2020

indexed features have to be used later to compare the similarity between the example image and the images in the indexed dataset, Figure 4 illustrates the process of indexing a dataset.

C. *Defining the similarity function*

The similarity function compares similarity features

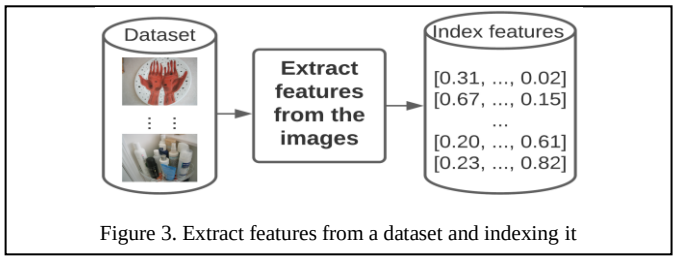

Figure 3. Extract features from a dataset and indexing it

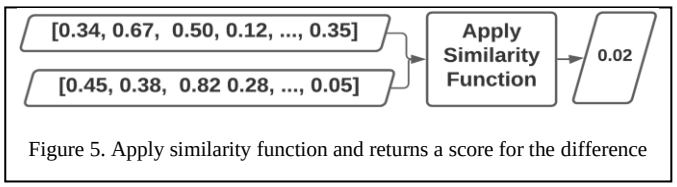

Figure 5. Apply similarity function and returns a score for the difference

between the example image with each image in the indexed dataset. The similarity function depends on Euclidean distance, Cosine distance, or chi-squared distance. However, the best choice is dependent on the used dataset, and the type of features have been extracted. The similarity function must return the range of deference. Figure 5 illustrates how to compare two images using the similarity function.

D. *Searching*

The last phase is to extract features from the example image and apply the suitable similarity function to search for the most similar feature (i.e., The less minimum deference distance) in the indexed dataset.

These are the most basic four phases used for any CBIR system. Firstly, a user must send a query image to the CBIR system. Then the system applies the descriptor on the query image to extract features from it. Where the system already indexed all images in the dataset. Then apply the similarity function to compare the "extracted query features" with the features of the images which already indexed in the dataset. Finally, the results are sorted by relevancy and presented to the user. Figure 4 illustrates the all phase process in a single figure.

III. DATASETS AND EVALUATION

For any CBVIR system, it is essential to choose an image dataset to validate and evaluate the accuracy of CBVIR system. An effective evaluate methods must be applied to estimate the accuracy such as precision, recall, f-score, and standard deviation. This section investigates some of the most used datasets then discover some metrics for evaluation. Also, it focuses on the importance of the environment in which the system works.

ImageNet is a Large-Scale Hierarchical high-resolution Image Database created by Deng in [3] includes over 15 million images belongs to 22,000 categories ingathered over the world wide web and labeled by humans using Amazon's Mechanical Truck tools.

UCID: Schaefer in [25] Proposed an Uncompressed Colour Image Database (UCID), which consists of 1338 uncompressed images on a variety of topics such as natural scenes and human-made objects, both indoors and outdoors.

Oxford: [20] The dataset contains 5062 images collected by searching for Oxford landmarks over Flickr website. Photographed images indicate eleven landmarks which are manually labeled. Five possible queries represent each landmark. That makes an object retrieval system could be evaluated over a group of 55 queries.

Paris is a dataset similar to Oxford dataset where it collects 6,412 for Paris landmarks from Flickr retrieval systems also could be evaluated over 55 queries.

Caltech256 proposed by Griffin in [7] contains 256 object categories includes 30607images which collected over google images. The range number of images in each category is from 30 to 80 images.

Pubfig83LFW: is mix of PubFig and LFW face datasets to produce an open-universe face identification image dataset. This dataset includes 13,002 faces representing 83 individuals from PubFig83 dataset.

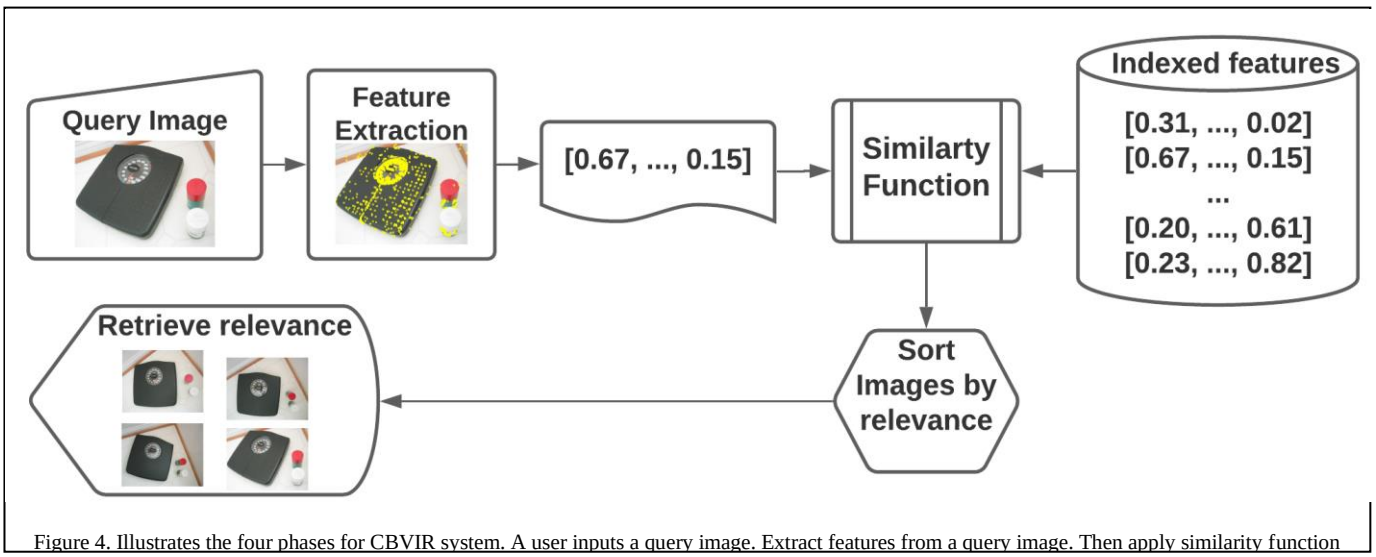

Figure 4. Illustrates the four phases for CBVIR system. A user inputs a query image. Extract features from a query image. Then apply similarity function

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 10, October-2020

University of Kentucky Benchmark: Nister in [18] proposed UKbench dataset, which consists of 10200 images divided into 2550 groups, where each group includes four images for the same object. The images are photographed from different points or in different light conditions. The sample Ukebench dataset contains only 1000 images divided into 225 groups. The dataset indexed in a JSON file where each image is a key for its four similar images, including the indexed image itself.

When measures CBIVR accuracy, images were considered as correct matches if the retrieved images belong to the same semantic class of the query image. Precision and recall are the most common evaluation measures in information retrieval, and those measures are used to evaluate the CBVIR system[19].

Precision is defined as the fraction of retrieved images in a result list that is relevant to a given query, and it is defined as:

$$p = \frac{\text{Relevant Results} \bigcap \text{Retrieved Results}}{\text{Retrieved Results}} = \frac{IR}{IT}$$

IR = Number of Relevance Images Retrieved.
IT = Total Number of Images Retrieved on the screen.
Recall (R) is defined as the fraction of documents that are relevant to the given query that are successfully retrieved. hence, measuring the ability of a system to present all relevant items. Recall is defined as:

$$R = \frac{\text{Relevant Results} \bigcap \text{Retrieved Results}}{\text{Relevant Results}} = \frac{IR}{I\,RB}$$

IR = Number of relevance Images Retrieved.
IRB =Total no of relevant images in the database.
Although both measures give a good indication of system performance, they are insufficient if they are just considered alone. A system can achieve higher recall by providing larger output to the user. The system which achieves higher recall may have low precision. On the other hand, higher precision can be achieved by providing fewer top-ranked images if the system has high early precision. This system will achieve higher precision but with lower recall. Some users early prefer precision, while others search for more relevant ones. Therefore, systems always try to balance between these two. A Precision-Recall curve is used to demonstrate system behavior concerning both precision and recall.

F-score could combine both precision and recall, sometimes called the F1-score or f-measure; it is defined as:

$$\text{F-Score}=2\frac{P.R}{P+R}$$

There are three standered evaluation metrics for retrieval systems (1) The Precision at particular ranks (p@k) as an example, P@10 indicates the number of relevant results in the first search page. However, the bad thing is that it cannot determine the position of the relevant result at the top k. (2) The recall at particular ranks (R@k). (3) The Mean Average Precision (MAP) which is the mean of the average precision scores for each query. Where the Average Precision AveP is:

$$\text{AvgP}(q)=\frac{1}{N_R}\sum_{n=1}^{N_R} P_Q(R_n)$$

Where R_n is the recall after nth relevant retrieved and N_R is the total number of relevant results for the query. So, MAP could be computed as:

$$\text{MAP}=\frac{\sum_{q=1}^{Q} AvgP_Q(R_n)}{Q}$$

Where Q is the number of queries.
Evaluation CBVIR system not only depends on the accuracy of retrieved images. The evaluation must take care of the speediness, dataset size, number of classes in the dataset, and the required space for saving the indexed feature. Sometimes the required space to store features of the dataset is larger than the dataset itself.

A. *Take care of the environment*

The quality of light in a given environment is crucial in obtaining CBVIR system goals. If constraints on the given environment could be controlled as the lighting position of the camera, contrast, and point of view, that will make the system more robust. The system ever developed will depend on the quality of images input to the system. However, not always photographed environment could be constrainted, hence the system may require more complex processes.

IV. CBVIR APPROACHES

Choosing the methodology or features descriptor technique depends on the dataset. Not always all successes systems give excellent results on all datasets. This section starts with a simple CBVIR system then has a look at BOVW approach at the end briefly presents some results on CBVIR which use deep learn/machine learning.

A. *CBVIR Systems via Color Histograms*

The most basic CBVIR system depended on color histogram. Histograms are used to provide a rough sense of the pixel intensity density in a picture. It is vital to select the number of bins for the histogram descriptors carefully. With few bins, as shown in figure 7, the histogram will have fewer Components that are incapable of differentiating between images with significantly different color distributions. Also, if it was a large number of bins, as shown in figure 8, There will be many components in the histogram and images with very similar content can be presented as not similar.

The small dataset should use a smaller number of bins to the histogram, i.e., with a larger dataset of images that uses more bins to make histogram larger. So, It in need to tune with the number of bins for color histogram descriptor as it is dependent on (1) the size of the dataset, (2) how similar the color distributions in the dataset are to each other. The histograms may be applied in many regions of the image to make the histogram more creative. Figure 9 show an image divided into five regions.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 10, October-2020

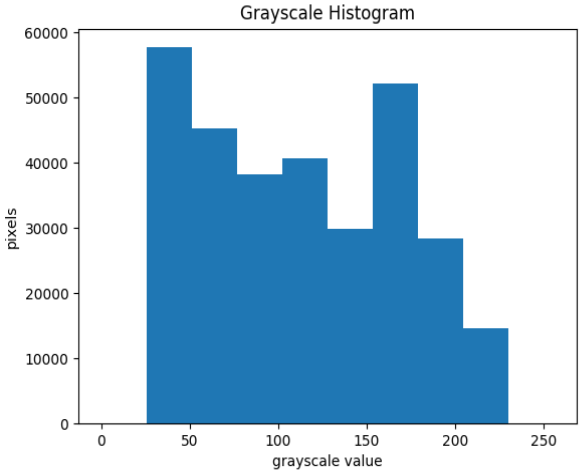

Figure 7. A 9-bin histogram example

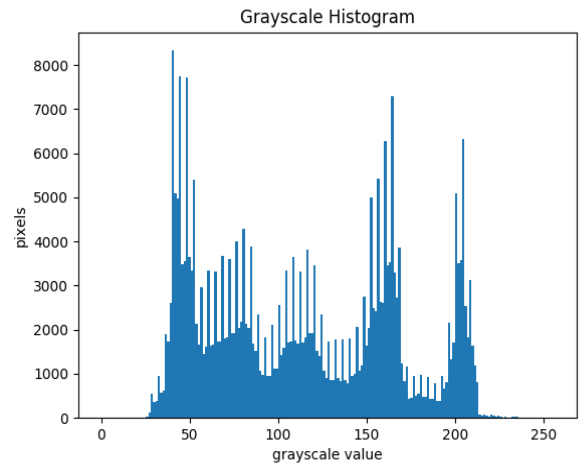

Figure 8. A 192-bin histogram.

After applying a histogram on all images in the dataset and the query example image, a similarity function should be defined to search or compare features of the example image with features of each image in the dataset. The chi-squared function is an excellent choice for color histogram systems.

$$d=0.5\sum_{i=1}^{n=1}\frac{a_i-b_i}{a_i+b_i+eps}\ for(a_i,b_i)\in(histA,histB)$$

Where:
d: represent the similarity between the two images A and B, The smaller d is, the two images are similar.
n: is the number of bins in each histogram histA and histB.
a: is the value of bin i in histA.
b: is the value of bin i in histA.
eps: is a very small value prevents dividing by zero.

Figure 5. Dividing an image into 5 different segments

The system often stores the data as a CSV file or as HDF5 [4]. Each line represents a feature vector of an image that leads to the searching phase requires loops over each line to compare with the query feature vector, so the search phase is O(N).

After the technique experimented on a simple UKbench dataset, the time required for indexing 1000 images is 12.3 seconds. Using (4,6,3) bins for HSV color channel histogram produces a feature vector of dimension 72. Using points evaluation as Nistér and Stewénius suggested in [18]. The accuracy is 3.34 of 4 (the summation of all points divided by the number of images in the dataset) or 0.91%, which are the standard deviation of all points. That gives excellent results with a small dataset, but with larger datasets, time of indexing and searching will be increase linearly and tuning the number of bins become overwhelmed. So turning to BOVW and Deep Learning/Machine Learning methods is an excellent choice to give better performance with larger datasets.

B. CBVIR via BOVW

Bag of Visual Words (BOVW) approach has a vast area in computer vision, such as CBVIR and image classification. this approach applies three main steps.

- Detect keypoints and extract features for each image in the dataset.
- Clustering all extracted features forming a vocabulary book of victor N.
- Describing all images in the dataset using a vocabulary book to construct Bag of visual words.

Ming-Kuei Hu's in [8] presented seven moments that characterize the shape of an object in an image. These moments are invariant to change in affine, rotation scaling, translation. So E.G. Karakasis in [9] used image moment invariant as local features for CBVIR using the BOVW approach for indexing and retrieval. The novelty here is that the user used Affine Moment Invariants (AMIs) to describe patches returned from the SURF detection mechanism. The experiments were conducted on UCID and UKbench datasets of images over four different codebook sizes (32, 128, 512, 2048) along with the three most commonly used weighting factors ($tf_{t.d}$, df_t, *normalization*), Where eight independent affine moment invariants are selected. In the result for every local feature and each codebook size, the evaluation was evaluated for all eight weighting Schemes (WS), but the

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 10, October-2020

greatest result only has been listed. On UCID [25] dataset with using SURF detection give rate 0.6513 for mean average precision (MAP) and 0.2088 for precision at particular rank k=10 (p@10) evaluation metric. This metric describes the system's capability to retrieve how many relevant imaged appears in the top 10 ranking, which is the first result page.

Also, experiments were evaluated on the UKbench [18] dataset using the N-S score. Where the retrieval score calculated by 4 × recall at the first four images in the result, so the maximum score is 4. At conducting experiments on the UKbench database, the best result for MAP evaluation also is 0.6513 and 2.4436 for the N-S score.

From the results of all the author experiments, it is noted that the small length of the codebook vector (32, 128) is not proper descriptors for the BOVW model on the used datasets. Shaping the codebook is an overwhelmed and challenge where, on the one hand, small codebook size leads to speed performance, but mostly it has less ability for distinguishing. On the other hand, lager codebooks size increases complexity cost and may produce inaccurate results. The codebook size depends on the number of clusters, which was an ambiguous procedure and mainly depends on the size and the type of image database.

C. CBVIR via Machine Learning and Deep learning

Some techniques that have been used to minimize the semantic gap are the exploration of image content using supervised learning to define object ontology for image labeling using machine learning methods such as logistic regression, Support Vector Machine (SVM), decision trees, random forests, and others to link low-level features with high-level semantics [13]. Image labeling/classification has been proposed as a preprocessing step to speed retrieval and classification systems [2]. Oppositely unsupervised learning has been introduced to increase performance when data are annotated or not labelled [14]. Machine learning is more related to image classification, where the core of the task is to assign a label to an image from a pre-defined set of categories. CBVIR via machine learning must take care of some of the challenges such as viewpoint variation, scale variation, deformation, occlusions, background clutter, and intra-class variation [17]. To overcome these challenges, Try to make the problem more narrow, considering the scope of the image classifier. There are three primary machine learning techniques the first is the supervised learning, which is the task of learning a function that maps an input to an output based on example input-output pairs. The favorite supervised learning includes Support Vector Machine (SVM), logistic regression, decision tree, and random forest. While the second is the unsupervised learning, which is a type of self-organized learning that helps to find patterns that were previously unknown in the dataset without pre-indexed labels. In unsupervised learning, there are no labels associated with the data points. Such unsupervised algorithms of machine learning arrange the data into a cluster category to explain its structure and make complex data look simple and ordered for analysis as K-means clustering algorithm. Lastly, semi-supervised learning which is a mix of supervised and unsupervised learning and usually is not accurate as supervised learning.

Deep learning is a method of artificial intelligence that mimics the functioning of the human brain in processing data and generating patterns for using in decision making. Deep learning is a subset of machine learning that has networks capable of learning from unstructured or unlabeled data. Deep learning contains groups of machine learning algorithms and methods that make systems able to learn complex functions. In Deep learning, many layers of data processing levels are in a hierarchical structure for pattern classification, and feature representation includes many modules such as neural networks, pattern recognition, graphical modeling, signal processing, and optimization. Recently deep learning methods have gained much interest in computer vision and machine learning [27]. These methods have shown increased effectiveness and efficiency in classification, recognition, and retrieval tasks, many deep learning methods attempt to model high-level abstractions in data [6].

Krizhevsky in [11] proposed the deep convolutional neural networks (CNNs) for image classification tasks with top-5 test error rate 15.3% using 1,000 classes. Then next researchers improved the models to achieve better results where reduced the error rate to be 13.24%. Deep solutions can offer higher accuracies, but with a large number of (balanced) image datasets, they need extensive training [12].

Wan in [27] proposed a deep learning framework using Convolutional Neural networks (CNNs) in two stages to improve the accuracy of results for CBVIR systems. The first stage is the training model, but the other stage is applying the trained deep model for learning feature representation of CBVIR on other data. The framework improved feature representation and Distance Metric Learning (DML). The author repeated the experiments on five image datasets ImageNet, Caltech256, Oxford, Paris, and Pubfig83LFW datasets. The author's results on ImageNet are more valuable than using the BoW technique in all states where the best MAP value for BoW is 0.0016, which is less than the smallest value for Deep framework MAP= 0.0748. Also, the results of experiments on the reset datasets in the worst state of the used deep framework are better than the BoW model.

Babenko in [1] used CNNs with seven layers to apply a content-based image retrieval system via neural network principles. The system has experimented on four different benchmark datasets includes Oxford, Oxford 105K, INRIA Holidays, and Ukbench datasets. The MAP values for the experimented datasets are 0.557, 0.522, 0.789, and 3.557, respectively, in the best cases. It is not good practice to compare results of a dataset with results of other datasets because each dataset has its particular case, such as type of objects in an image, number of classes, number of similar images, lighting condition, and point of view.

It is not doubt that the convolution between the advantages of different approaches produces a new better method. The two primary keys effect on any CBIR system are the similarity measures and feature representation, which could be improved by applying CNNs and SVM algorithms. Ruigang Fu in [5] has built a CBVIR system based on CNN and SVM techniques. Where the author used SVM learning for similarity measures on pre-trained CNN experimented evaluated on Caltech256 dataset on different class sizes

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 10, October-2020

(10,20,50) with MAP values 0.7976, 0.6182, and 0.4281 respectively.

V. CONCLUSION

In recent two decades, many research papers have been published to overcome the problem of the semantic gap between low-level pixel representation in machines and high-level concepts interpreted by a human for the content of an image over the CBVIR domain. The two main factors that reduce the semantic gap are the features representation and distance similarity matrices. Many methods, such as color histogram features, BOVW, and machine learning algorithms, developed to improve the performance of the two factors. From the experiments noted that color histogram descriptor gives particularly satisfying results over narrows datasets. Using BOVW, invariant methods decrease the time of searching, but via deep learning and machine learning techniques, minimum search time with accurate results over large scale datasets appeared. Tuning parameters such as the number of bins for a color histogram or number of clusters to form the codebook for BOVW requires many experiments to be well determined. It is an excellent idea to combine the pros of different techniques to build content-Based visual information retrieval via multi-features fusion.

REFERENCE

[1] Babenko, A., et al., "Neural codes for image retrieval," *in European conference on computer vision*, pp. 584-599, 2014.

[2] Carneiro, G., et al., "Supervised learning of semantic classes for image annotation and retrieval," *IEEE transactions on pattern analysis and machine intelligence*, vol. **29**, no. 3, pp. 394-410, 2007.

[3] Deng, J., et al., "Imagenet: A large-scale hierarchical image database," *in 2009 IEEE conference on computer vision and pattern recognition*, pp. 248-255, 2009.

[4] Folk, M., et al., "An overview of the HDF5 technology suite and its applications," *in Proceedings of the EDBT/ICDT 2011 Workshop on Array Databases*, pp. 36-47, 2011.

[5] Fu, R., et al., "Content-based image retrieval based on CNN and SVM," *in 2016 2nd IEEE International Conference on Computer and Communications (ICCC)*, pp. 638-642, 2016.

[6] Gordo, A., et al., "Deep image retrieval: Learning global representations for image search," *in European conference on computer vision*, pp. 241-257, 2016.

[7] Griffin, G., A. Holub, and P. Perona, "Caltech-256 object category dataset," 2007.

[8] Hu, M.-K., "Visual pattern recognition by moment invariants," *IRE transactions on information theory*, vol. **8**, no. 2, pp. 179-187, 1962.

[9] Karakasis, E.G., et al., "Image moment invariants as local features for content based image retrieval using the bag-of-visual-words model," *Pattern Recognition Letters*, vol. **55**, pp. 22-27, 2015.

[10] Kato, T., "Database architecture for content-based image retrieval," *in image storage and retrieval systems*, pp. 112-123, 1992.

[11] Krizhevsky, A., I. Sutskever, and G.E. Hinton, "Imagenet classification with deep convolutional neural networks," *in Advances in neural information processing systems*, pp. 1097-1105, 2012.

[12] Kumar, M.D., et al., "A comparative study of CNN, BoVW and LBP for classification of histopathological images," *in 2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, pp. 1-7, 2017.

[13] Laaksonen, J., M. Koskela, and E. Oja, "PicSOM-self-organizing image retrieval with MPEG-7 content descriptors," *IEEE Transactions on Neural Networks*, vol. **13**, no. 4, pp. 841-853, 2002.

[14] Le, Q.V., "Building high-level features using large scale unsupervised learning," *in 2013 IEEE international conference on acoustics, speech and signal processing*, pp. 8595-8598, 2013.

[15] Liu, Y., et al., "A survey of content-based image retrieval with high-level semantics," *Pattern recognition*, vol. **40**, no. 1, pp. 262-282, 2007.

[16] Madaan, G., "Various Approaches of Content Based Image Retrieval Process: A Review," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. **3**, no. 1, 2018.

[17] Nilsback, M.-E. and A. Zisserman, "A visual vocabulary for flower classification," *in 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, pp. 1447-1454, 2006.

[18] Nister, D. and H. Stewenius, "Scalable recognition with a vocabulary tree," *in 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, pp. 2161-2168, 2006.

[19] Pattanaik, S. and D. Bhalke, "Beginners to Content-Based Image Retrieval," *International Journal of Science, Engineering and Technology Research*, vol. **1**, pp. 40-44, 2012.

[20] Philbin, J., et al., "Object retrieval with large vocabularies and fast spatial matching," *in 2007 IEEE conference on computer vision and pattern recognition*, pp. 1-8, 2007.

[21] Piras, L. and G. Giacinto, "Information fusion in content based image retrieval: A comprehensive overview," *Information Fusion*, vol. **37**, pp. 50-60, 2017.

[22] Purbey, A., M. Sharma, and B. Bohra, "Review on: Content Based Image Retrieval," 2017.

[23] Rui, Y., et al., "Relevance feedback: a power tool for interactive content-based image retrieval," *IEEE Transactions on circuits and systems for video technology*, vol. **8**, no. 5, pp. 644-655, 1998.

[24] Rui, Y., T.S. Huang, and S.-F. Chang, "Image retrieval: Current techniques, promising directions, and open issues," *Journal of visual communication and image representation*, vol. **10**, no. 1, pp. 39-62, 1999.

[25] Schaefer, G. and M. Stich, "UCID: An uncompressed color image database," *in Storage and Retrieval Methods and Applications for Multimedia 2004*, pp. 472-480, 2003.

[26] Smeulders, A.W., et al., "Content-based image retrieval at the end of the early years," *IEEE Transactions on pattern analysis and machine intelligence*, vol. **22**, no. 12, pp. 1349-1380, 2000.

[27] Wan, J., et al., "Deep learning for content-based image retrieval: A comprehensive study," *in Proceedings of the 22nd ACM international conference on Multimedia*, pp. 157-166, 2014.

[28] Wang, B., et al., "A semantic description for content-based image retrieval," *in 2008 International Conference on Machine Learning and Cybernetics*, pp. 2466-2469, 2008.

[29] Wang, R., et al., "A novel method for image classification based on bag of visual words," *Journal of Visual Communication and Image Representation*, vol. **40**, pp. 24-33, 2016.