10 (1) January 2024 60-73 ISSN 2502-3357 (online) | ISSN 2503-0477 (print)
Register: *Jurnal Ilmiah Teknologi Sistem Informasi*

Contents lists available at www.journal.unipdu.ac.id

Register

Journal Page is available to www.journal.unipdu.ac.id/index.php/register

Research article

Content-Dependent Image Search System with Automatic Weighting Mechanism for Aggregating Color, Shape, and Texture Features

Arvita Agus Kurniasari [a*], *Ali Ridho Barakbah* [b], *Achmad Basuki* [c]

[a] *Department of Information Technology, Jember State Polytechnic, Jl. Mastrip, Sumbersari, Jember, Jawa Timur, 68121, Indonesia*
[b,c] *Informatic and Computer Department, Electronical Engineering Polytechnic Institute of Surabaya, Jl. Raya ITS, Sukolilo, Surabaya, Jawa Timur, 60111, Indonesia*
email: [a,*] arvita@polije.ac.id, [b] ridho@pens.ac.id, [c] basuki@pens.ac.id
* *Correspondence*

ARTICLE INFO

Article history:
Received 1 May 2023
Revised 8 October 2023
Accepted 4 April 2024
Available online 27 April 2024

Keywords:
Content-dependent image search;
Image retrieval;
Feature aggregation;
Automatic weighting.

Please cite this article in IEEE style as:
A. A. Kurniasari, A. R. Barakbah, A. Basuki, "Content-Dependent Image Search System with Automatic Weighting Mechanism for Aggregating Color, Shape, and Texture Features" Register: Jurnal Ilmiah Teknologi Sistem Informasi, vol. 10, no. 1, pp. 60-73, 2024.

ABSTRACT

The existing image search system extracts features from the database images and performs queries thoroughly without considering the weight of each feature. Currently, all features are assigned the same weight, even though each image has different characteristics. This study proposes a new approach to image search systems that relies on content with automatic weighting. The automatic weighting process starts by calculating each moment. The first moment is obtained from the color matrix and is calculated as the average value. The second moment is obtained from the texture matrix and is calculated as the variance value. The third moment is obtained from the shape matrix and is calculated as the skewness value. These three moments are normalized to give the same weight to each feature for each picture. The results obtained for accuracy were: 70.38% for color, 60.99% for shape, 71.21% for texture, 72.65% for color-shape combinations, 78.43% for color-texture combinations, 72.65% for texture-shape combinations, and 80.5% for overall texture-color-shape features.

1. Introduction

The advancement of technology through digital information is accelerating due to globalization, which reduces geographical, spatial, and temporal constraints. This progress results in the proliferation of data, making it harder for consumers to choose and get high-quality information. The accuracy, timeliness, and relevance of knowledge contribute to its quality [1]. Accuracy implies error-free and objective information that effectively serves its intended purpose, as changing information can disrupt the intended message. Therefore, information accuracy is crucial in the search process [2]. Search engines are commonly used to retrieve information, and efforts are continuously made to enhance their functionality, including the development of image search algorithms. These algorithms aim to achieve the highest possible accuracy [3]. Numerous studies have explored image search systems. For example, a study was concerned with object detection in natural photographs within unconstrained domains, aiming to gather crucial information for efficient image capture, based on user semantic judgments at a visual level [4]. The primary challenge in developing this technology is accurately segmenting images into distinct sections. An inhomogeneous vector diffusion model was developed using many colors, textures, and form properties to address these issues. This approach is robust and successful in generating area-based image descriptions, but it also helps optimize item segmentation [5]. Another

ISSN 2502-3357 (online) | ISSN 2503-0477 (print)
10 (1) January 2024 60-73

study focuses on picture search with an analytical function that considers shape, structure, and color characteristics. It quantizes three-dimensional color vectors to extract color features and utilizes metrics like eccentricity, area, diameter, and convex area for shape characteristics. A curvelet is employed to isolate structural elements. Test results demonstrate the system's effectiveness in enhancing accuracy, unaffected by company-specific characteristics [6]. The results from another study show that a weighted feature selection technique outperforms the K-means strategy in terms of annotation performance and correspondence accuracy. This technique assigns and updates weights to individual dimensions throughout the learning process. The goal is to connect visual objects and low-level characteristics with ideas (keywords). This research not only examines images but also considers language; the weighting assigned to each image corresponds to the associated keywords [7].

The application of image search in large database applications is not trivial [8]. Numerous studies have been conducted, each with its own set of advantages and disadvantages. This research aims to integrate several factors to achieve the highest possible accuracy outcomes through Content-Based Image Retrieval (CBIR). CBIR is a content-based image search technique that focuses on the characteristics of a set of images [9]. The fundamental premise of CBIR is to use image analysis algorithms to automatically extract subsets of picture attributes from an image database one at a time using image analysis techniques. These attributes may include numeric values indicating hues, textures, and shapes. CBIR can enhance the accuracy and efficiency of the search process and manage large amounts of image data [10]. CBIR searches for photographs that meet specific image criteria from a group of existing images based on image information. The final image's characteristics or criteria include shapes, colors, textures, and others that correspond to the desired image [11][12].

The current image search technology lacks the ability to perform content-based searches [13]. The investigation is carried out comprehensively, and it processes the image promptly, without considering the image's content. The investigation is carried out wholly and promptly processes the image without regard for the image's content. For instance, in a photograph of a mountain scene with mountains, clouds, and roads, the current algorithm extracts all the information from the image without recognizing that the mountains are the main focus of the landscape image's content. As a result, the retrieved photographs may include unrelated elements such as street views or landscapes with clouds, which are not as expected. Another issue arises during feature extraction, where the features used focus only on one or two aspects rather than integrating all of them.

Meanwhile, not all photos analyzed provide adequate information for a single type of feature extraction, as in the case of a white cloud image. If the extraction process is limited to color characteristics, the output may include a white wall image along with the texture and shape of the print, which is distinct from the cloud picture. This article introduces a novel technique for content-dependent image search systems based on an object identification mechanism and automated feature selection weighting. It utilizes color, shape, and texture as characteristics. The initial processing stage involves extracting RGB and HSL values from the color characteristics. The extracted data is normalized before being used. Clustering is also used to remove objects from a picture. This method aims to accurately capture the image's content, ensuring that the system precisely captures the image's content. An automated weighting mechanism will be introduced to the feature extraction process to ensure that the features selected for the final approach match the features included in the picture. The results will be ranked based on the similarity between the images in the input query and the database, thus ensuring higher accuracy.

2. Materials and Methods

The image data used in this study is the Wang et al.'s benchmark 1000-image Simplicity dataset [14], which serves as both input and query data. This dataset is divided into ten categories, including images of Africans, buildings, beaches, buses, dinosaurs, flowers, horses, elephants, and mountains. The system modeling technique comprises five major steps: preprocessing, clustering, object identification,

 https://doi.org/10.26594/register.v10i1.3501

ISSN 2502-3357 (online) | ISSN 2503-0477 (print)
10 (1) January 2024 60-73

feature extraction, and similarity assessment. Subsequently, the search results are retrieved. Fig 1 illustrates the system design.

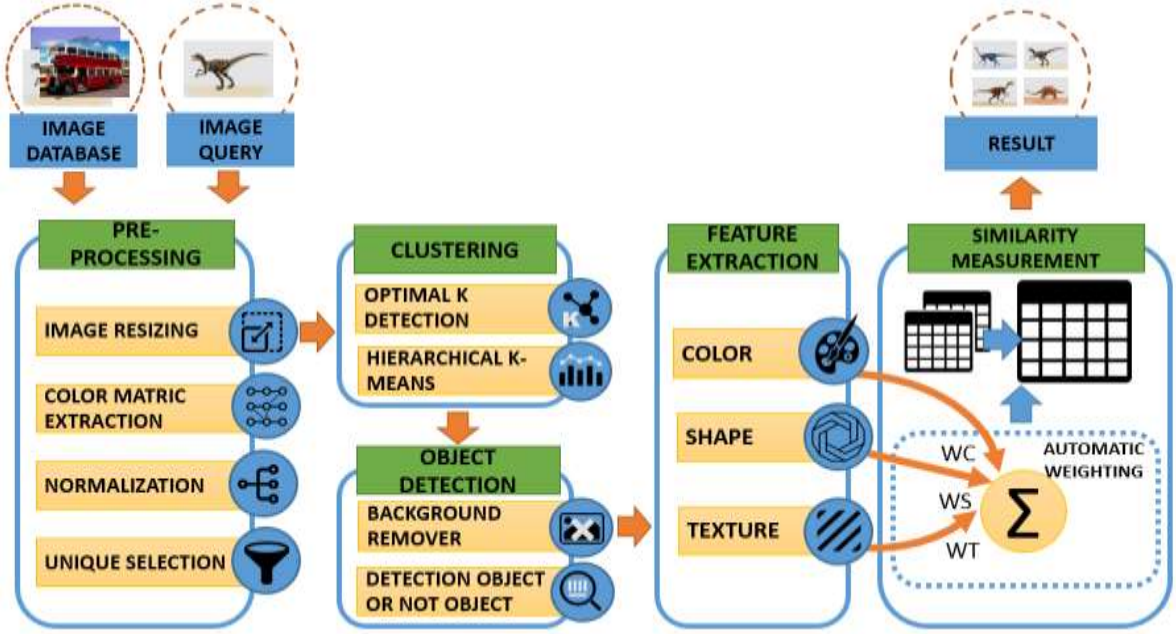


Fig. 1. System Diagram

The system is constructed in five steps, starting with preprocessing. In this step, the image is resized from a large size to a smaller size, such as 20 x 30 pixels. This is achieved by extracting RGB and HSL values from the image, normalizing the extraction results into metadata using softmax, and removing unique values from the color metric normalization results. These stages generate two distinct color value measurements, one for large sizes and one for small sizes. The second stage is clustering, which involves determining the number of groups using an automated clustering method based on the color metric values of small-sized images. The significant image color value metric is employed, and the approach used is Hierarchical K-means, determined by the number of clusters in the small image findings.

The third step is object detection. In this stage, the background is removed from the image, and objects are extracted using the Hierarchical K-means clustering results. The system then assesses whether the background removal process led to a reduction in the number of objects. It calculates the object-to-background ratio to determine this. Non-objects are processed differently, without background removal. Meanwhile, non-objects are processed differently, without background removal. The fourth stage is feature extraction, where color, shape, and texture characteristics are extracted. The color feature takes advantage of the 3D-Color Vector Quantization technology, which removes the color information from the poll picture and serves as a color value throughout the rendering process. The form feature utilizes the Linked Component Labeling approach, which converts binary images to symbolic representations and assigns a unique label to each connected component. Texture characteristics are extracted using the Leung Malik approach, which involves using a filter bank.

The next stage is similarity measurement, which involves computing the normalized Canberra distance between the query image and the images in the database and then ranking them based on the acquired space. Before calculating the distance, an automatic weighting approach is applied, particularly for automatically weighting the outcomes of color, shape, and texture feature extraction. This stage is completed to ensure that each feature is assigned a weight corresponding to its respective component, avoiding unequal weighting between elements. This paper aims to develop a comparable image search application based on Android that utilizes color-based object recognition in input images. The authors intend to evaluate the accuracy of the detected objects and the technique used to extract relevant images. The following procedures were followed.

 https://doi.org/10.26594/register.v10i1.3501

ISSN 2502-3357 (online) | ISSN 2503-0477 (print)
10 (1) January 2024 60-73

2.1. Preprocessing

The initial processing stage begins by duplicating the image and reducing its size to 20x30 pixels, while maintaining the original image resolution at 256x384 pixels. The rationale behind the conversion to 20x30 is to reduce clustering time. Researchers experimented with different ratios to determine the optimal one, and the 20x30 ratio proved to be the most effective in terms of speed and accuracy. After the reduction, the color characteristics of the two pictures are extracted; the values extracted are the position values (x and y), RGB, and HSL. The HSL value is determined by the outcome of the RGB value conversion. The HSL value is derived from the RGB value conversion, as HSL is known to enhance the HSV color system by more accurately representing brightness than saturation. The resulting data are normalized using the SoftMax technique, which converts the position, RGB, and HSL to average 0–1 range values. Finally, the data is sorted for unique values to expedite the initial processing.

2.2. Clustering

Clustering was performed once the image data was converted into a color metric. This stage begins by generating a color matrix for small images for the initial clustering process using Get Optimal K. The data clustering stage determines the optimal number of clusters for presenting data metrics by identifying possible positions to determine the variants' optimal global movement [15] [16]. When performing significant picture clustering using Hierarchical 3K-means [17], the optimal global value of this tiny image color matrix is employed as a reference. For instance, the ideal number of clusters based on the small picture matrix is five. After that, the number 5 is utilized as a reference for calculating the number of enormous picture matrix groups. By displaying the number of clusters of tiny pictures, this clustering seeks to segment images from an extensive image color matrix. Since clustering huge photos directly would be time-consuming and demand significant storage space. While the segmentation results are not flawless, the approach is quite effective at displaying the content of the selected image.

2.3. Object Detection

The object detection procedure was conducted in two stages [13]: background removal and object identification. The background removal stage begins with a clustered image matrix constructed using the tiny image matrix technique, as shown in Fig 2 with five clusters. The segmentation clustering findings are then used to delete the backdrop on four sides, including the top left, upper right, bottom left, and lower right. The color matrix on the left side is examined for the values of the dominant cluster. The cluster information from the color matrix is consolidated into a single set, with the value replaced by 1 to indicate the presence of an object. Conversely, the value for the non-dominant cluster is set to 0. The same procedure is applied to the other sides, setting the color matrix to 1 for dominant clusters and 0 for non-dominant clusters. The final step is to determine the ratio of objects to the background. For example, if the clustering results indicate a higher inner balance ratio compared to the outer ratio, it suggests the presence of an object. If the clustering results show a higher outer ratio, then the object is not present. In this scenario, the article considers a ratio of 25%–75% of the image area as indicating the presence of an object. If objects are detected, only the clustering values are processed to the next step. If no objects are detected, only the overall value results are processed further.

2.4. Feature Extraction

One of the most critical challenges in image search systems is selecting feature extraction methods that align with the measurement approach. The features extracted in this study include color, shape, and texture. The authors employed 3D-Color Vector Quantization and HSV for the color extraction process. The 3D-Color Vector Quantization technique operates on the principle that the system consistently displays the color of an image at any point in the RGB vector color space. This process simplifies the RGB colors in the image and aligns them with the vector space [18]. The quantization size is set to 5X5X5, resulting in 125 RGB positions. A histogram is then created by locating and organizing the location values of each pixel.

 https://doi.org/10.26594/register.v10i1.3501

ISSN 2502-3357 (online) | ISSN 2503-0477 (print)
10 (1) January 2024 60-73

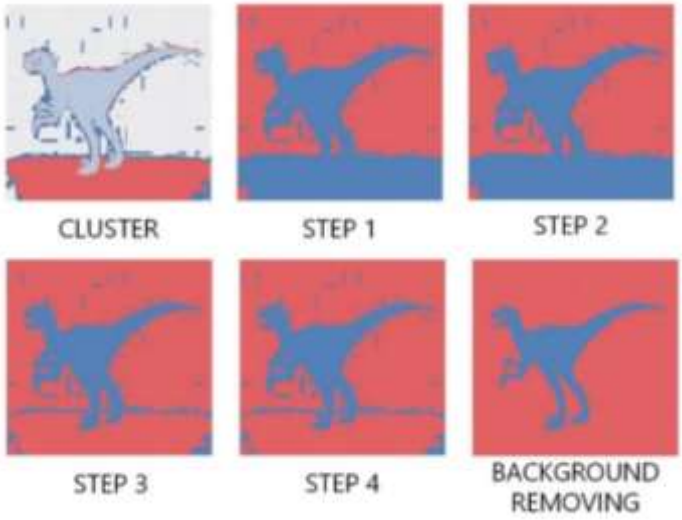
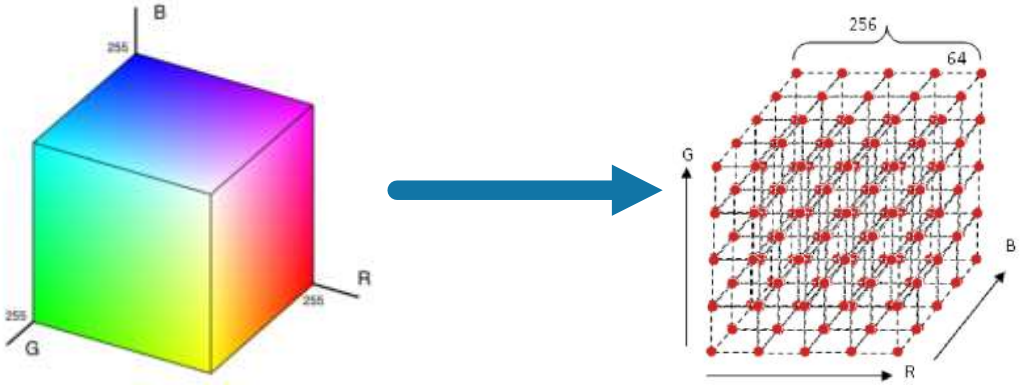


Fig. 2. Implementation of background removing



(Source : *A.R.Barakbah, Y.Kivoki, 2008, 3D-Color Vector Quantization for Image retrievalSistem, international database Sistem (IDB) 2008, Izaka, Japan* [19])

Fig. 3. Illustration of 3D-Color Vector Quantization RGB

HSV (hue, saturation, value) colors are commonly used because they closely resemble the way humans perceive color. The HSV color space is represented in three dimensions by a hexagon, with the central vertical axis representing an intensity range of 0 to 255 [20]. The construction of the histogram is similar to that of the RGB color space in HSV image search. Each histogram section is updated for each pixel with the correct color combination [21]. Effective histogram indexing may significantly increase the performance of CBIR [22]. Color features extracted using the HSV histogram include hue values ranging from 0 to 179, saturation values ranging from 0 to 255, and value values ranging from 0 to 1. The HSV histogram is used to calculate the process of color feature extraction. The final computation is made by aggregating HSV values to create a histogram consisting of 122 values: 60 H values ranging from 0 to 179, 60 S values ranging from 0 to 255, and 2 for V, indicating a value of 0 for black and 1 for white. The following formula is utilized to extract color features:

$$MC_{x,i} = \{fc_{x,1}, fc_{x,2}, fc_{x,3}, \dots, fc_{x,125}\} \quad (1)$$

Here, $fc_{x,i}$ represents the extracted features from an i color histogram.

The author's feature extraction technique utilizes Connected Component Labeling (CCL) [23], which is employed based on the thresholds of various models to identify all groups—a geographically linked group of pixels with the same pixel intensity value or related to one another. Thus, each pixel is then labeled with the component's name once all groups have been formed. Edge detection utilizing canny detection [24] is used to smooth out images and noise. Subsequently, contours from objects are extracted using Connected Component Labeling (CCL). Each object's area, equivalent diameter, extension, convex hull, solidity, eccentricity, and perimeter are calculated using the contours. Calculations are performed to convert each pixel to the author's shape matrix using the mean (μ), median, standard deviation (σ), variance (σ^2), skewness (s), and kurtosis (k).

 https://doi.org/10.26594/register.v10i1.3501

ISSN 2502-3357 (online) | ISSN 2503-0477 (print)
10 (1) January 2024 60-73

$$\mu_i = \frac{1}{N}\sum_{j=1}^{N} X_{ij} \quad (2)$$

$$\sigma_i = \left(\frac{1}{N}\sum_{j=1}^{N}(X_{ij} - \mu_i)^2\right)^{\frac{1}{2}} \quad (3)$$

$$\sigma_i = \frac{1}{N}\sum_{j=1}^{N}(X_{ij} - \mu_i)^2 \quad (4)$$

$$s_i = \frac{1}{N}\sum_{j=1}^{N}\left[\frac{X_{ij}-\mu_i}{\sigma_i}\right]^3 \quad (5)$$

$$k_i = \frac{1}{N}\sum_{j=1}^{N}\left[\frac{X_{ij}-\mu_i}{\sigma_i}\right]^4 \quad (6)$$

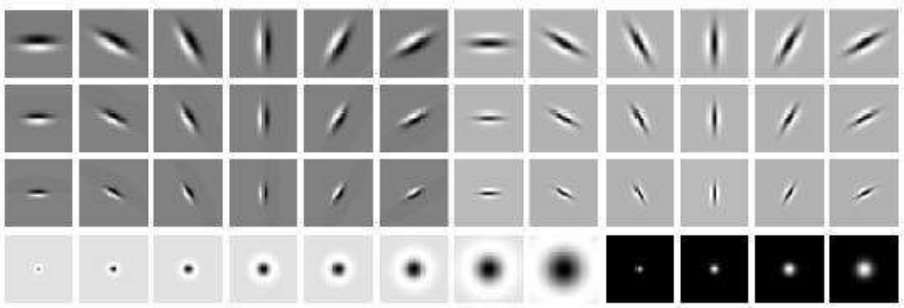
Here, X_{ij} denotes the value of the color component *i* in the j picture pixel, and N is the image pixel count.

The edge detection findings are utilized in conjunction with the Connected Component Labeling (CCL) approach to generate the object's contours. From these contours, metrics such as area, equivalent diameter, extent, convex hull, solidity, eccentricity, and perimeter of the object or non-object are determined. The first image in the first row and column demonstrates the contour results of the CCL method. The second graphic in the first row and second column depicts the area, equivalent diameter, and extent of the resulting contours after calculation. The third picture in the second row and first column shows the convex hull and solidity results from that row. The fourth picture in the second row and second column displays the results of the eccentricity calculation. The last images in the third row and first column illustrate the results of the perimeter calculation. The following formula is utilized to extract shape features:

$$MS_{x,i} = \{fS_{x,area}, fS_{x,equv}, fS_{x,ext}, fS_{x,conv}, fS_{x,sol}, fS_{x,ecc}, fS_{x,per}\} \quad (7)$$

$fS_{x,area}$ is a result of feature extraction from an *area* in block x;
$fS_{x,equv}$ is a result of feature extraction from the *equivalent diameter* in block x;
$fS_{x,ext}$ is a result of feature extraction from *extent* in block x;
$fS_{x,conv}$ is a result of feature extraction from the *convex hull* in block x;
$fS_{x,sol}$ is a result of feature extraction from *solidity* in block x;
$fS_{x,ecc}$ is a result of feature extraction from *eccentricity* in block x;
$fS_{x,per}$ is a result of feature extraction from the *perimeter* in block x.

The author's texture was extracted using the Leung Malik transformation (LM). The rotating variation of the Leung Malik (LM) filter bank is the LM filter bank. When the LM filter bank is rotated, the derivatives will change. The LM filter bank considers a collection of photos acquired under a predefined set of imaging parameters as the input data [25]. The LM filter set, shown in Fig 4, consists of 48 filters: the first two Gaussian derivatives in six directions and three scales, four Gaussian filters, and eight Laplacian of Gaussian filters [26]. The Leung Malik kernel is implemented using a deft edge segmentation algorithm [27]. The authors' texture extraction method uses the mean (μ), median, standard deviation (σ), variance (σ^2), skewness (s), and kurtosis(k).



(Source: www.robots.ox.ac.uk/~vgg/research/texclass/filters.html)

Fig. 4. The Leung Malik filter bank

The following formula is utilized to extract texture features:

$$MT_{x,i} = \{fT_{x,1}, fT_{x,2}, fT_{x,3}, \ldots, fT_{x,15}\} \quad (8)$$

Here, $fT_{x,1}$ represents a result of texture feature extraction from *i* kernel.

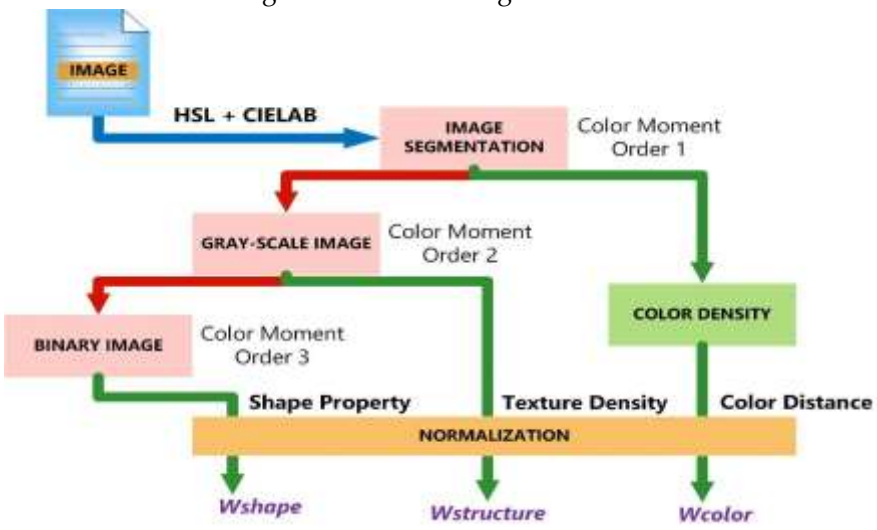
2.5. Automatic Weighting

This work is distinct from other image searches as it incorporates automated weighting steps after feature extraction and before similarity measurement stages. These stages are conducted to determine the feature weights used; hence, not all characteristics are utilized, but the feature weights for each

 https://doi.org/10.26594/register.v10i1.3501

ISSN 2502-3357 (online) | ISSN 2503-0477 (print)
10 (1) January 2024 60-73

image are determined. The initial stage involves keeping track of every second. This calculates the image's color moment by computing the color distance and texture density for color and shape weights. Many search methods have effectively used color features, which are efficient and effective for describing the distribution of color pictures [28]. The first moment is calculated using the color matrix, which is used to calculate the mean (μ) value; the second moment is calculated using the texture matrix, which is used to calculate the variance (σ^2) value; and the third moment is calculated using the form matrix, which is used to calculate the skewness (s) value. These three moments are standardized to ensure that each feature in each image has the same weight.



(Source: *A. R. Barakbah and Y. Kiyoki, "Image Search System with Automatic Weighting Mechanism for Selecting Features," 6th Int. Conf. Inf. Commun. Technol. Syst., 2010* [28])

Fig. 5. Auto weighting illustration

2.6. Similarity Measurement

Similarity measurement stages are conducted based on the extraction results. The researchers calculate the Canberra distance and its normalization between query images and the image database. They also use Cosine for spaces and sort the obtained distances. Finally, the results are displayed and analyzed. The Canberra distance is a modification of the Manhattan distance, which relates the difference in attribute values to the values themselves.

$$D_{(x,y)} = \sum_i^N \frac{|x_i - y_i|}{|x_i| + |y_i|} \quad (9)$$

Here, N represents the number of attributes; x_i indicates metadata from query images; and y_i represents Metadata from the dataset image.

The Normalized Canberra distance normalizes the distance data for each attribute [29] by comparing two data points between query images and image data. Here is the equation for the Normalized Canberra Distance:

$$D_{(x,y)} = \frac{\sum_i^N \frac{|x_i - y_i|}{|x_i| + |y_i|}}{N} \quad (10)$$

Here, N represents the number of attributes; x_i indicates metadata from query images, and y_i is Metadata from the dataset image.

The Cosine Distance considering two vectors, X and Y, where X = $(x_1, x_2, \dots, x_n)$ and Y = $(y_1, y_2, \dots, y_n)$, then cos θ can be regarded as the Cosine of the angle between vector X and Y in dimensions [30].

$$D_{(x,y)} = \frac{\sum_{i=1}^N x_i . y_i}{\sqrt{\sum_{i=1}^N {x_i}^2 . \sum_{i=1}^N {y_i}^2}} \quad (11)$$

Here, N represents the number of attributes; $\boldsymbol{x_i}$ is metadata from query images; and $\boldsymbol{y_i}$ refers to Metadata from the dataset image.

 https://doi.org/10.26594/register.v10i1.3501

ISSN 2502-3357 (online) | ISSN 2503-0477 (print)
10 (1) January 2024 60-73

3. Results and Discussion

3.1. Color Feature Method Selection

The color feature selection procedure begins with a comparison between 3D-Color Quantization and HSV histograms, followed by a comparison of both accuracy techniques using the k-nearest neighbor algorithm (k-NN) with k = 3. Based on the experimental findings of the researchers, the optimal value of K was determined to be an odd number, specifically three, which provided the most accurate result. The overall color matrix is then combined into a single matrix, and the error, accuracy, and score values are calculated. The Euclidean distance equation for the k-nearest neighbor algorithm (k-NN) technique is as follows:

$$D_{(x,y)} = \sqrt{\sum_i^N (x_i - y_i)^2} \quad (12)$$

Here, N represents the number of attributes; x_i is metadata from query images; and y_i is Metadata from the dataset image.

$$Error = \sum_{i=1}^{10} err_i \begin{cases} err_i = 0 \leftarrow cr_i = cq \\ err_i = 1 \leftarrow otherwise \end{cases} \quad (13)$$

$$Score = \sum_{i=1}^{10} scr_i \begin{cases} scr_i = 10 - i + 1 \leftarrow cr_i = cq \\ scr_i = 0 \leftarrow otherwise \end{cases} \quad (14)$$

$$Precision = \sum_{i=1}^{10} pre_i \begin{cases} pre_i = 1 \leftarrow cr_i = cq \\ pre_i = 0 \leftarrow otherwise \end{cases} \quad (15)$$

Here, cr_i denotes the category of retrieval image, and cq is category from image query.

Table 1. Comparison of the Results from Color Feature Methods

Images	Error		Accuracy		Score	
	3D-Color Vector Quantization	HSV Histogram	3D-Color Vector Quantization	HSV Histogram	3D-Color Vector Quantization	HSV Histogram
African People	0.676	0.572	0.324	0.428	21.742	27.443
Beach	0.668	0.672	0.332	0.328	21.928	21.876
Building	0.744	0.522	0.256	0.478	18.960	30.475
Bus	0.713	0.566	0.287	0.434	20.076	27.707
Dinosaur	0.135	0.191	0.865	0.809	49.444	47.343
Elephant	0.653	0.618	0.347	0.382	23.561	25.204
Flower	0.554	0.449	0.446	0.551	29.227	34.196
Food	0.549	0.576	0.451	0.424	28.813	27.177
Horse	0.375	0.247	0.625	0.753	37.968	43.495
Mountain	0.727	0.599	0.273	0.401	19.323	25.475
Average	0.579	**0.501**	0.421	**0.499**	27.104	**31.039**

As demonstrated in Table 1, adopting the HSV Histogram approach for color feature extraction outperforms the 3D-Color Vector Quantization method. From the accuracy data presented above, it is evident that using the HSV histogram with 122 values yielded a maximum accuracy of 0.499 percent.

3.2. Shape Feature Selection

The accuracy of the shape feature selection process is assessed using the k-nearest neighbor algorithm (k-NN) with k = 3. This evaluation is based on the mean (μ), median, standard deviation (σ), variance (σ^2), skewness (s), and kurtosis (k) values calculated for the contour implementation's area, equivalent diameter, extent, convex hull, solidity, eccentricity, and perimeter values. As shown in Table 2, utilizing seven parameters—specifically the area value, equivalent diameter, extension, convex hull, solidity, eccentricity, and perimeter—is more advantageous than using only five parameters without extending or solidifying the object. Based on the accuracy statistics provided, it is evident that employing softmax normalization and calculating the average, standard deviation, and median results in a maximum accuracy of 60.99 percent.

3.3. Texture Feature Selection

The accuracy of the k-nearest neighbor's algorithm (k-NN) with k = 3 is determined by comparing the mean (μ), median, standard deviation (σ), variance (σ^2), skewness (s), and kurtosis(k) of the 15 Leung Malik kernel implementations (LM).

 https://doi.org/10.26594/register.v10i1.3501

ISSN 2502-3357 (online) | ISSN 2503-0477 (print)
10 (1) January 2024 60-73

Tabel 2. Calculation Results for Shape Feature Parameter Accuracy

Calculation	5 parameter			7 parameter		
	sigmoid	softmax	without normalize	sigmoid	softmax	without normalize
Mean	44.48	48.61	46.75	50.15	56.04	48.71
mean_stdv	47.47	51.91	49.12	53.04	56.97	45.2
mean_stdv_median	52.32	56.35	48.19	52.32	**60.99**	51.7
mean_stdv_skew	45.82	51.08	48.61	52.32	56.86	48.92
mean_stdv_kur	51.08	56.86	48.4	49.54	57.17	47.68
mean_stdv_skew_kur	48.3	51.81	48.71	52.01	56.66	48.19
mean_stdv_skew_kur_median	53.66	54.28	47.99	50.88	56.35	45.61
mean_stdv_median_skew	48.71	53.87	51.08	52.12	57.17	51.91
mean_stdv_median_kur	54.8	55.42	48.92	53.66	56.35	45.51
mean_varian	44.17	55.11	46.23	49.95	55.31	41.9
mean_varian_median	51.19	57.38	46.23	50.05	57.69	42.21
mean_varian_skew	45.3	51.19	46.34	53.56	55.01	41.8
mean_varian_kur	51.29	57.79	46.13	51.81	56.66	47.27
mean_varian_skew_kur	48.19	51.5	46.13	50.26	55.93	47.27
mean_varian_skew_kur_median	53.25	54.9	46.13	51.7	57.79	47.88
mean_varian_median_skew	47.06	53.77	46.34	51.81	58.62	42.11
mean_varian_median_kur	53.56	55.73	46.23	52.43	59.24	47.88
All	52.73	54.39	46.13	50.98	57.38	47.88
Average	49.6	54.0	47.4	51.6	**57.1**	46.6

Tabel 3. Calculation Results for Texture Feature Parameter Accuracy

Calculation	sobel (1-6,19-24,43,44,45)			canny (1-6,19-24,43,44,45)		
	sigmoid	softmax	without normalize	sigmoid	softmax	without normalize
Mean	50.1	64.7	66.1	56.9	63.8	63.9
mean_stdv	52.0	66.2	63.7	53.4	69.3	66.8
mean_stdv_median	52.5	67.1	66.1	56.2	70.2	68.5
mean_stdv_skew	54.8	66.4	63.4	60.2	**71.2**	66.9
mean_stdv_kur	49.9	64.5	63.8	59.9	68.5	69.6
mean_stdv_skew_kur	52.5	67.9	63.4	59.3	69.4	65.5
mean_stdv_skew_kur_median	54.8	67.3	65.9	59.9	69.1	67.2
mean_stdv_median_skew	55.0	64.6	65.3	60.8	66.8	69.1
mean_stdv_median_kur	49.5	65.2	65.7	58.6	69.0	71.2
mean_varian	50.3	64.7	60.2	56.8	69.7	64.6
mean_varian_median	53.8	66.7	60.8	55.4	70.0	66.1
mean_varian_skew	56.6	65.7	60.1	59.7	71.1	64.6
mean_varian_kur	50.6	66.4	60.7	58.2	70.6	65.1
mean_varian_skew_kur	53.6	66.1	61.0	59.3	69.4	65.5
mean_varian_skew_kur_median	54.3	67.5	61.7	59.9	69.1	67.2
mean_varian_median_skew	53.6	66.8	60.9	59.0	68.9	66.2
mean_varian_median_kur	54.2	67.3	61.5	58.5	69.5	67.3
All	54.6	64.2	61.4	60.8	71.1	67.3
Average	52.9	66.1	62.9	58.5	**69.3**	66.8

As seen in Table 3, the implementation of canny edge detection outperforms Sobel edge detection. Based on the accuracy results presented, it is evident that combining softmax normalization with the calculation of the average value, standard deviation, and skewness leads to a maximum accuracy of 71.2 percent.

3.4. Comparison of Automatic Weighting Techniques

Automatic weighting testing techniques are carried out by directly comparing the matrix of each overall feature with the weighted feature matrix. The combined matrix of colors, shapes, and textures is then calculated, followed by the computation of the error, accuracy, and score value. The weighted feature

 https://doi.org/10.26594/register.v10i1.3501

ISSN 2502-3357 (online) | ISSN 2503-0477 (print)
10 (1) January 2024 60-73

matrix means that the weight of each feature is calculated, then multiplied by its extraction value, and the total value is summed.

Table 4. Results of Weighting Comparisons and Manual Weighting of Features

Images	Error		Accuracy		Score	
	Manual	Automatic	Manual	Automatic	Manual	Automatic
African people	0.45	0.48	0.54	0.51	34.33	32.45
beach	0.58	0.55	0.42	0.44	27.22	28.01
building	0.42	0.43	0.57	0.56	35.15	34.3
bus	0.21	0.22	0.78	0.77	46.52	45.7
dinosaur	0.07	0.07	0.92	0.92	51.68	51.52
elephant	0.41	0.43	0.58	0.56	35.98	35.48
flower	0.18	0.17	0.81	0.82	47.01	47.27
food	0.37	0.41	0.62	0.59	37.94	36.17
horse	0.22	0.21	0.77	0.78	45.31	45.43
Average	**0.34**	0.35	**0.65**	0.64	**39.24**	38.6

According to Table 4, the application of automatic weighting is not as effective as manual weighting. This is due to suboptimal form feature extraction techniques, which consequently affect the results of the weighting calculations.

3.5. Selection of Similarity Measurement Techniques

In selecting similarity measurement techniques, the researchers have compared the normalized Canberra distance, Canberra distance, and cosine distance on the calculated feature merges, with each component weighted accordingly.

Table 5. Similarity Measurement Cosine Comparison Calculation Results

Images	Cosine-Normalized Canberra			Cosine- Canberra		
	Error	Accuracy	Score	Error	Accuracy	Score
African People	0.93	0.07	3.47	0.58	0.42	27.22
Beach	0.71	0.29	17.85	0.61	0.39	25.58
Building	0.58	0.42	25.67	0.64	0.36	24.13
Bus	0.48	0.52	30.16	0.25	0.75	44.15
Dinosaur	0.15	0.85	47.95	0.23	0.77	44.53
Elephant	0.97	0.03	1.74	0.45	0.55	32.70
Flower	0.32	0.68	41.10	0.41	0.59	35.66
Food	0.91	0.09	5.70	0.53	0.47	29.77
Horse	0.63	0.37	21.01	0.44	0.56	34.81
Mountain	0.70	0.30	18.73	0.73	0.27	18.98
Average	0.64	0.36	21.35	0.49	0.51	31.67

Tables 5 and 6 present error results, precision, and scores from comparing different approach methods. The Normalized Canberra distance approach and the Canberra distance method are more effectively applied to the combined process of color, shape, and texture feature extraction when automatic weighting is used, compared to a technique that calculates cosine distance with automatic weighting.

Table 6. Similarity Measurement Cosine Comparison Calculation Results

Images	Normalized Canberra			Canberra		
	Error	Accuracy	Score	Error	Accuracy	Score
Africa People	0.52	0.48	30.41	0.48	0.52	32.45
Beach	0.59	0.41	26.61	0.56	0.44	28.01
Building	0.54	0.46	30.21	0.43	0.57	34.30
Bus	0.27	0.73	43.59	0.23	0.77	45.70
Dinosaurs	0.14	0.86	48.69	0.08	0.92	51.52
Elephant	0.36	0.64	38.13	0.44	0.56	35.48
Flower	0.26	0.74	43.32	0.18	0.82	47.27
Food	0.43	0.57	34.83	0.41	0.59	36.17
Horse	0.29	0.71	42.61	0.22	0.78	45.43
Mountain	0.64	0.36	24.07	0.52	0.48	30.16
Average	0.40	0.60	36.19	**0.36**	**0.64**	**38.60**

 https://doi.org/10.26594/register.v10i1.3501

ISSN 2502-3357 (online) | ISSN 2503-0477 (print)
10 (1) January 2024 60-73

3.6. Analysis of Image Search Results

Comparing each feature plays an essential role in understanding the results of the image search approach, whether by combining all features or using only one component. The testing involves comparing each feature–color , shape, and texture–using k-nearest neighbor's algorithm method calculation k-NN with k = 3.

Table 7. Results of Calculating Average Error Ratios, Precision, and Scores

	Accuracy
Color	70.38
Texture	71.21
Shape	60,99
Color-Texture	78.43
Color-Shape	72.65
Texture-Shape	72.65
Color-Texture-Shape	**80.5**

Table 7 shows the highest accuracy of 80.5% obtained by combining the features of color, shape, and texture. Conversely, the lowest accuracy, at 60.99%, is obtained when using the shape feature alone. Therefore, it can be concluded that combining all the components yields better results than using only one. Fig 6 (a), Fig 6 (b), and Fig 6 (c) present the error ratios, precision, and scores calculated for each category. The dinosaur category shows the highest average precision and score among the features compared.

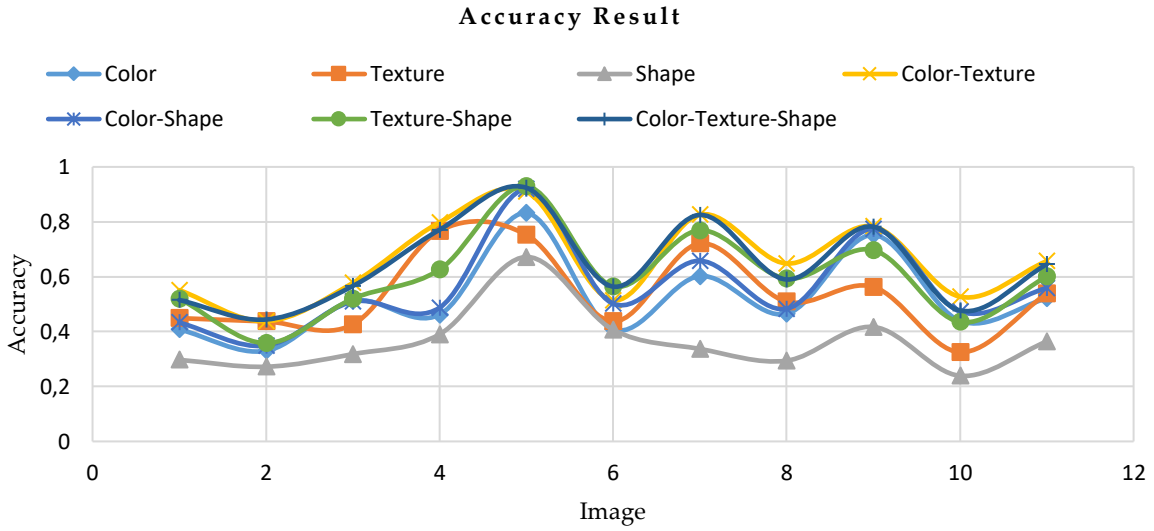


Fig. 7. Accuracy Result (b) error ratio

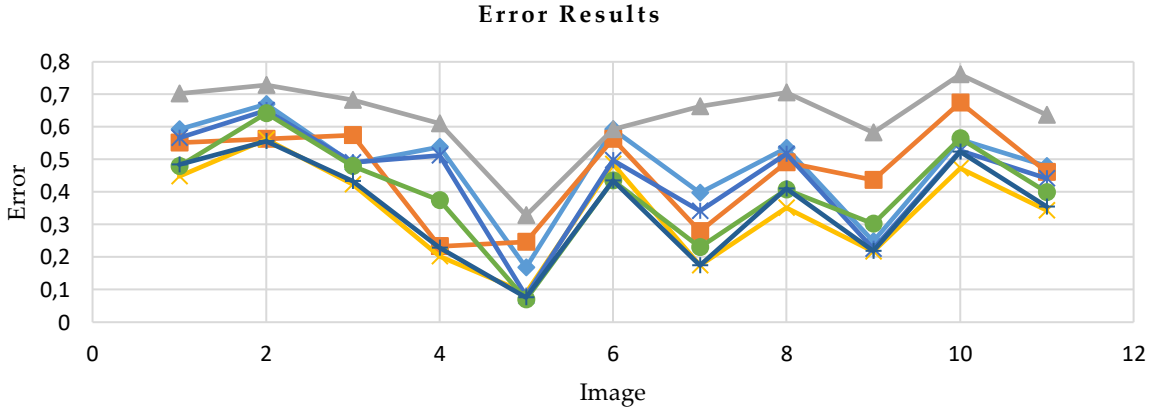


Fig. 8. Error Ratio Result

https://doi.org/10.26594/register.v10i1.3501

ISSN 2502-3357 (online) | ISSN 2503-0477 (print)
10 (1) January 2024 60-73

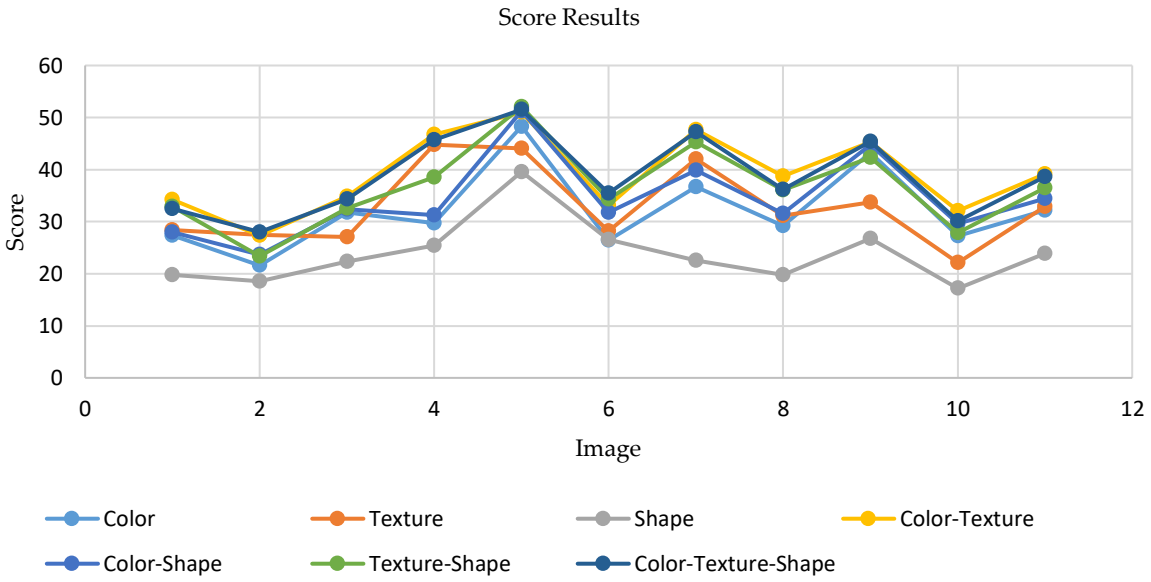


Fig. 9. Score Result

In contrast, the beach category shows the lowest average precision and score among the features compared. This difference in results can be attributed to the fact that the dinosaur category contains background information that is less variable compared to the beach category. The dinosaur category provides more relevant information than background objects when objects are detected. The results of the dinosaur category approach are superior to those of the beach category because the information captured is more relevant. The technique used to capture images represents a weakness in this research that can affect the results. The researchers opted not to use original images due to differences in preprocessing stages.

4. Conclusion

This research focuses on the content-dependent image search system with an automatic weighting mechanism for aggregate color, shape, and texture features. The goal is to facilitate the system in selecting elements that match the weight of each component, aiming for more optimal results. The existing system includes preprocessing stages that utilezes RGB and HSL color compositions, employing the Hierarchical K-means and Optimal K Selection method to identify objects from images. Feature extraction is applied using features of color, shape, and texture. For color features, the researchers used HSV histogram; for shape extraction features, Connected Component Labeling (CCL) was employed, and for texture features, Leung Malik's kernel was used. This paper introduces an approach that uses automatic weighting to approximate the results of manual weight calculation. The comparison between the normalized Canberra distance and the Canberra distance calculation shows that they are more effectively applied to the combined approach of color, shape, and texture feature extraction with automatic weighting, compared to a method involving cosine distance calculation. The obtained results for accuracy are as follows: color accuracy of 70.38%, shape accuracy of 60.99%, texture accuracy of 71.21%, color-shape combination accuracy of 72.65%, color-texture combination accuracy 78.43%, texture-shape combination accuracy of 72.65%, and overall accuracy for texture-color-shape features of 80.5%.

Author Contributions

A. A. Kurniasari: Conceptualization, formal analysis, methodology, project administration, software, validation, visualization, writing – original draft, and writing – review & editing. A. R. Barakbah: Conceptualization, formal analysis, methodology, supervision, validation, and writing – review & editing. A. Basuki: Conceptualization, formal analysis, methodology, supervision, and validation.

Acknowledgment

The author would like to express his appreciation to the Director, the Center for Research and Community Service, the Department of Informatics and Computers at the Jember State Polytechnic, and

ISSN 2502-3357 (online) | ISSN 2503-0477 (print)
10 (1) January 2024 60-73

the Department of Informatics and Computers at the Electronics Polytechnic Surabaya Institute, Indonesia, for sponsorship of research.

Declaration of Competing Interest

We declare that we have no conflict of interest.

References

[1] M. I. Jaya, F. Sidi, I. Ishak, L. S. Affendey, and M. A. Jabar, "A review of data quality research in achieving high data quality within organization," *J. Theor. Appl. Inf. Technol.*, vol. 95, no. 12, pp. 2647–2657, 2017.

[2] G. Pennycook, Z. Epstein, M. Mosleh, A. A. Arechar, D. Eckles, and D. G. Rand, "Shifting attention to accuracy can reduce misinformation online," *Nature*, vol. 592, no. 7855, pp. 590–595, 2021, doi: 10.1038/s41586-021-03344-2.

[3] A. . F. Adrakatti, R. S. Wodeyar, and K. R. Mulla, "Search by Image: A Novel Approach to Content Based Image Retrieval System," *Int. J. Libr. Sci.*, vol. 14, no. 3, pp. 41–47, 2016, [Online]. Available: http://www.ceser.in/ceserp/index.php/ijls/article/view/4561

[4] S. Joseph and O. O. Olugbara, "Detecting salient image objects using color histogram clustering for region granularity," *J. Imaging*, vol. 7, no. 9, 2021, doi: 10.3390/jimaging7090187.

[5] F. Saba, M. J. Valadan Zoej, and M. Mokhtarzade, "Optimization of Multiresolution Segmentation for Object-Oriented Road Detection from High-Resolution Images," *Can. J. Remote Sens.*, vol. 42, no. 2, pp. 75–84, 2016, doi: 10.1080/07038992.2016.1160770.

[6] A. R. Barakbah and Y. Kiyoki, "Image Retrieval Systems with 3D-Color Vector Quantization and Cluster based Shape and Structure Features," *Inf. Model. Knowl. Bases XXI*, vol. 206, pp. 169–187, 2010.

[7] A. Al-Mohamade, O. Bchir, and M. M. Ben Ismail, "Multiple query content-based image retrieval using relevance feature weight learning," *J. Imaging*, vol. 6, no. 1, 2020, doi: 10.3390/jimaging6010002.

[8] F. H. D. Araujo *et al.*, "Reverse image search for scientific data within and beyond the visible spectrum," *Expert Syst. Appl.*, vol. 109, pp. 35–48, 2018, doi: 10.1016/j.eswa.2018.05.015.

[9] I. M. Hameed, S. H. Abdulhussain, and B. M. Mahmmod, "Content-based image retrieval: A review of recent trends," *Cogent Eng.*, vol. 8, no. 1, 2021, doi: 10.1080/23311916.2021.1927469.

[10] A. Kumar, S. Choudhary, V. S. Khokhar, V. Meena, and C. Chattopadhyay, "Automatic Feature Weight Determination using Indexing and Pseudo-Relevance Feedback for Multi-feature Content-Based Image Retrieval," pp. 1–9, 2018, [Online]. Available: http://arxiv.org/abs/1812.04215

[11] A. Latif *et al.*, "Content-based image retrieval and feature extraction: A comprehensive review," *Math. Probl. Eng.*, vol. 2019, 2019, doi: 10.1155/2019/9658350.

[12] R. C. Winedhar, "Komputasi Budaya Untuk Pencarian Gambar Semantik Pada Lukisan Budaya Indonesia Dengan Deteksi Dan Informasi Aliran Lukisan," *J. Teknol. Inf. dan Terap.*, vol. 8, no. 1, pp. 6–12, 2021, doi: 10.25047/jtit.v8i1.224.

[13] A. A. Kurniasari, A. R. Barakbah, and A. Basuki, "Content-Dependent Image Search System for Aggregation of Color, Shape and Texture Features," *Emit. Int. J. Eng. Technol.*, vol. 7, no. 1, pp. 223–242, 2019, doi: 10.24003/emitter.v7i1.361.

[14] G. W. Jia Li James Ze Wang, "SIMPLIcity: Semantics-sensitive Integrated Matching for Picture LIbraries," *IEEE Pami*, vol. 23, no. 9, pp. 947–963, 2001, doi: 10.1109/34.955109.

[15] M. Maleki, N. Manshouri, and T. Kayıkçıoğlu, "A Novel Simple Method to Select Optimal k in k-Nearest Neighbor Classifier," vol. 15, no. 2, pp. 464–469, 2017.

[16] A. R. Barakbah and K. Arai, "Determining Constraints of Moving Variance to Find Global Optimum and Make Automatic Clustering," *Ind. Electron. Semin. 2004*, pp. 409–413, 2004.

[17] J. Qi, Y. Yu, L. Wang, J. Liu, and Y. Wang, "An effective and efficient hierarchical K-means clustering algorithm," *Int. J. Distrib. Sens. Networks*, vol. 13, no. 8, pp. 1–17, 2017, doi: 10.1177/1550147717728627.

[18] A. R. Barakbah and Y. Kiyoki, "IMAGE RETRIEVAL SYSTEMS WITH 3D-COLOR VECTOR QUANTIZATION AND CLUSTER BASED SHAPE AND STRUCTURE FEATURE

ISSN 2502-3357 (online) | ISSN 2503-0477 (print)
10 (1) January 2024 60-73

EXTRACTION System Design Color Feature Extraction Shape & Structure Feature Extraction," p. 1000.

[19] A. R. Barakbah and Y. Kiyoki, "3D-Color Vector Quantization for Image Retrieval Systems," *Int. Database Forum 2008*, no. September 2008, pp. 13–18, 2008.

[20] M. A. Sayeed, "Detecting Crows on Sowed Crop Fields using Simplistic Image processing Techniques by Open CV in comparison with TensorFlow Image Detection API," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 8, no. 3, pp. 61–73, 2020, doi: 10.22214/ijraset.2020.3014.

[21] T. D. Pupitasari *et al.*, "Intelligent detection of rice leaf diseases based on histogram color and closing morphological," *Emirates J. Food Agric.*, vol. 34, no. 5, pp. 404–410, 2022, doi: 10.9755/ejfa.2022.v34.i5.2858.

[22] H. Qazanfari, H. Hassanpour, and K. Qazanfari, "Content-Based Image Retrieval Using HSV Color Space Features," *Int. J. Comput. Inf. Eng.*, vol. 13, no. 10, pp. 537–545, 2019.

[23] L. He, X. Ren, Q. Gao, X. Zhao, B. Yao, and Y. Chao, "The connected-component labeling problem: A review of state-of-the-art algorithms," *Pattern Recognit.*, vol. 70, pp. 25–43, 2017, doi: 10.1016/j.patcog.2017.04.018.

[24] M. A. Ansari, D. Kurchaniya, and M. Dixit, "A Comprehensive Analysis of Image Edge Detection Techniques," *Int. J. Multimed. Ubiquitous Eng.*, vol. 12, no. 11, pp. 1–12, 2017, doi: 10.14257/ijmue.2017.12.11.01.

[25] S. Basheera and M. Satya Sai Ram, "Alzheimer's disease classification using leung-malik filtered bank features and weak classifier," *Int. J. Recent Technol. Eng.*, vol. 8, no. 3, pp. 1956–1961, 2019, doi: 10.35940/ijrte.C4484.098319.

[26] E. S. Varnousfaderani, S. Yousefi, C. Bowd, A. Belghith, and M. H. Goldbaum, "Vessel Delineation in Retinal Images using Leung-Malik filters and Two Levels Hierarchical Learning," *AMIA ... Annu. Symp. proceedings. AMIA Symp.*, vol. 2015, pp. 1140–1147, 2015.

[27] A. Sengur, Y. Guo, M. Ustundag, and Ö. F. Alcin, "A Novel Edge Detection Algorithm Based on Texture Feature Coding," *J. Intell. Syst.*, vol. 24, no. 2, pp. 235–248, 2015, doi: 10.1515/jisys-2014-0075.

[28] A. R. Barakbah and Y. Kiyoki, "Image Search System with Automatic Weighting Mechanism for Selecting Features," *6th Int. Conf. Inf. Commun. Technol. Syst.*, 2010.

[29] M. Faisal, E. M. Zamzami, and Sutarman, "Comparative Analysis of Inter-Centroid K-Means Performance using Euclidean Distance, Canberra Distance and Manhattan Distance," *J. Phys. Conf. Ser.*, vol. 1566, no. 1, 2020, doi: 10.1088/1742-6596/1566/1/012112.

[30] P. dwi Nurfadila, A. P. Wibawa, I. A. E. Zaeni, and A. Nafalski, "Journal Classification Using Cosine Similarity Method on Title and Abstract with Frequency-Based Stopword Removal ," *Int. J. Artif. Intell. Res.*, vol. 3, no. 2, 2019, doi: 10.29099/ijair.v3i2.99.