International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

# Constraint and Descriptor Based Image Retrieval through Sketches with Data Retrieval using Reversible Data Hiding

**Dipika Birari *[1], Dilendra Hiran[2], Vaibhav Narawade[3]**

**Submitted:** 06/06/2022 **Accepted:** 10/09/2022

***Abstract:*** An image retrieval system includes image retrieval through sketches. Sketches act as an outline for any object with few details. "Sketch-Based Image Retrieval (SBIR)" is universally recognized as an extension of image retrieval by such rough sketching that concentrates on the main features of the object. SBIR has become an effective and popular image mining search technique as the demand for multimedia technology has grown. Due to the less precise depiction in sketches, comparing such sketches to real colorful and meaningful images becomes extremely difficult. As a solution to the captioned matter, the proposed approach incorporates Histogram Line Relationship (HLR) descriptors to facilitate constraint-based image retrieval. After pre-processing, the descriptor describes the visual features of an image. Here edge length-based constraints make SBIR powerful enough to select strong shaping edges. This approach is further enhanced to include data retrieval and is referred to as "Sketch-Based Image and Data Retrieval (SBIDR)" which even makes it more functional. Throughout image processing, the data embedding and extraction procedure are carried out using the Reversible Data Hiding (RDH) technique with an invariant grayscale version. The proposed method employs a hybrid model of image retrieval and data retrieval system with the addition of constraints and grayscale invariance. This models produce efficient outcomes in terms of retrieval.

***Keywords:*** *Image retrieval, Descriptor, Data retrieval, Edge extraction, Sketch-based, Grayscale, Invariance, feature extraction.*

## 1. Introduction

The extensive use of the internet and the growing demand for storage, multimedia data such as audio, video, and photos has prompted an evolution in multimedia retrieval systems. As a result, image retrieval has become a prominent tool for image processing in this era, and it is referred as widely used as a way of exploring images from huge databases. The text-based search was the very first phase in the evolution of image retrieval; this annotation-based technique searches the database based on the surrounding text of an image. As string matching processes are less time-consuming, this manual or automatic text annotation of images works perfectly. However, there are restrictions on how each image's contents are represented in the text. It may be impossible to constantly express image content in text or word, as image annotation for image retrieval is not always accurate.

Due to various changes in the retrieval mechanism, such as querying by image content, content-based search has evolved as an alternative to text-based search and is now widely used around the world. Instead of depending on "metadata," content-based search focuses on the internal content of images, hence it is referred to as “Content-Based Image Retrieval (CBIR)”. Hereby removing text annotation from visual content, querying through it becomes easier.

Internal features, or low-level picture features including colour, texture, shape, and spatial locations, are used to characterize CBIR. A feature database is a set of target picture feature vectors in which multidimensional feature vectors are generated based on the feature description. The similarity measure is used to calculate the distance between the target picture's feature vector in the vector database and the query image vector. Ultimately, image retrieval is accomplished through the use of an indexing technique.

Following that, a freehand sketch query is introduced, which is not a replacement for CBIR but can be used as an add-on. Sketch-Based Image Retrieval (SBIR) is an emerging trend in image search technologies nowadays. SBIR evolved and dramatically enhanced in the field of Image Processing, inspired by the well-known phrase "A picture is worth a thousand words”. Because pictures are more expressive than keywords, queries are provided as sketches to search for real images with in repository. In every sense, the freehand sketches enhance image searches. With this in mind, an extensive study is being conducted on SBIR and its applicability in law enforcement, military, and criminal justice investigations where a drop in operational quality may result in a crucial loss. SBIR, in addition to the CBIR, is a strong search tool that can be utilized as a supplement to text-based search. Pre-processing, followed by edge extraction descriptor generation, contour detection, constraint satisfaction, and perhaps other operations, helps this method succeed.

Grayscale invariance is critical when utilising sketches to search

[1] *Research Scholar, Faculty of Computer Engineering, Pacific Academy of Higher Education and Research University, Udaipur, Rajasthan, India*
*ORCID ID : 0000-0001-5767-6157*
[2] *2Pacific Institute of Computer Applications, Udaipur, Rajasthan, India*
[3] *Department of Computer Engineering Ramrao Adik Institute of Technology, Nerul, Navi Mumbai, Maharashtra, India*
** Corresponding Author Email: dipikabirari001@email.com*

for pictures since sketches are depicted with Black and White tones while other comparable images are multi-colored. Images are frequently used to depict imagination. Image processing is a cognitive process in which we perform alterations on pictures or use various processing techniques. During the alterations or during processing, due care should be taken to maintain the image's attributes. Since sketches are black and white while other comparable images are coloured, grayscale invariance is critical when utilising sketches to search for images. Because grayscale images have lower intensities than colour images, they can be processed faster. Data is incorporated into colour graphics, with red, green, and blue colour channels providing messages and sketches containing the key.

RDH [2] is frequently used on real images with this method to extract data hidden in the image. It's a fragile mechanism used to authenticate. Histogram shift, lossless compression, and expansion difference algorithms are used in several RDH approaches. This approach operates on the grayscale, and there are several algorithms available for conducting watermarking on colour photos such as the one shown. Colour images, on the other hand, offer more powerful visualisation than grey images, which are blue-channel based and three-color channel based [26]. However, these algorithms irrevocably damage their colour pictures, as a result a noisy image is generated which varies from the original image. Marked images intensify the sting extraction or feature extraction process. SIFT and HoG, for example, are feature descriptors for grey photos that only operate on the luminance channel after converting colour images to grey images. Certain RDH approaches for colour images may be utilised in SBIR colour images to avoid interfering with subsequent Gray-scale invariance image processing. The RDH is utilised to process Gray-scale invariance and retrieve data from images in this work, with tagged images being used for subsequent image retrieval via sketching.

The Sketch-Based Image and Data Retrieval (SBIDR) System is a hybrid image and data retrieval system shown in the following Fig. 1.

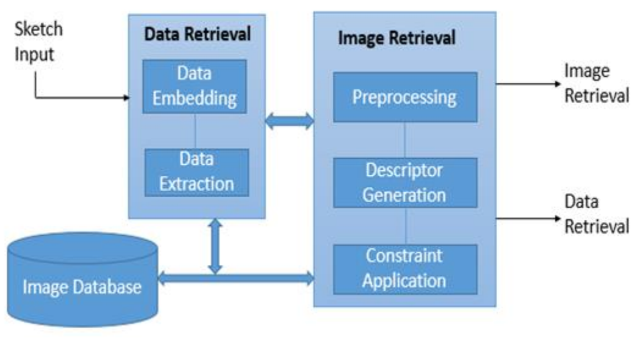



**Figure 1.** Image and Data Retrieval through Sketches

## 2. Related Work

A system that searches, navigates, and retrieves images from a huge digital images repository or text is known as an image retrieval system. The initial phase in the evolution of image retrieval is text-based image retrieval (TBIR), in which text or keywords play an important role in the retrieval system. It is solely based on verbal descriptions of images, and retrieval is entirely dependent on word annotation. Because of the limitations on feature description via keywords or text, finding the images just by metadata (annotations) is insufficient. CBIR is the next iteration of image retrieval that resulted from this. Images with compound backgrounds may include a variety of visual features; hence CBIR incorporates a number of feature descriptors to represent image visual features. Image alterations such as rotation, scaling, and luminance changes have no effect on the descriptor. Depending on extraction methods descriptors are classified. RBIR are local descriptors derived from an image portion, whereas GCBIR are global descriptors extracted from the overall visual content of the picture. These are very insightful and time-consuming techniques. SBIR evaluates images using query sketches, the most expressive and dynamic technique to image retrieval. The freehand picture of anything is a sketch that can be either an object or anything from one's mind. Real-life images feature numerous elements such as colour, shape, texture, gradient, scale, and so on, whereas sketches are instructive and useful for learning about shapes. CBIR is incapable of bridging such a wide gap between sketches and actual images.

J. Canny [4] created a canny detector for image pre-processing that reduced the volume of data in a query and target image without affecting the image's structural features. The procedure includes cleaning, identifying gradients and magnitude, double thresholding, edge suppression, and edge tracking. Another method of picture extraction is contour detection and extraction. Eitz et al. [5] provide a human perception approach for addressing the problem of matching a contour to a query sketch picture. To analyse an image, two sorts of regions are frequently used: Region of Interest (ROI) and primary region. ROI is useful when the backdrop of a photograph is complex, and it works on a significant section of the image. Eitz et al. introduced the Tensor Descriptor concept, which divides an image into cells, each of which serves as a descriptor. In Hierarchical Orientation Combination, the human visual system, which uses a hierarchical framework for picture processing, is exploited. The maximum RGB colour component is used to calculate orientation information. Because there is no index structure in this database, the algorithm does a comprehensive scan of the whole database for each probe.

Hu et al. [6] introduce the GF-HoG descriptor as an alternative method for extracting window based descriptors in order to improve and optimise performance. HoG distributes pictures into cells, which are linked to tiny portions termed regions, utilising a collection of visual words described in [7]. Each cell generates a histogram comprising gradient orientations and directions. The representation of cells in a histogram is normalised to generate a block histogram, and the collection of these block histograms is referred to as a descriptor. Object detection is performed using a window-based descriptor, and spatial information is necessary for improved accuracy. MindFinder[8] displays the permutation of edge pixel orientations and coordinates as a dictionary of spatial information and magnifies the resulting shape feature descriptor.

An image recovery technique called patch hashing [9] divides a picture into overlapping patches and computes a feature description called a Histogram of Gradients (HoG) for each recovered patch. This method builds a system for obtaining duplicate pictures from query drawings with slight location disparities, i.e. It looks for images with better spatial persistence. Hu and collomosse[10] developed GF-HOG descriptors that work on dense gradients with local properties determined by selecting an appropriate window size. Yang et al. [11] suggested a 12-dimensional feature vector to split contour into multiple components that are invariant to scaling and rotation. Although it has drawbacks such as high memory use, a Hungarian Algorithm is used to determine the similarity of two vectors. In order to reduce memory usage, Xiao et al. [14] divided the contour into separate shaping words that include shapes like curves and lines and measured the similarity of the shape words. The Chan Vese Model [12] expresses it in terms of mean intensities. It makes use of both the inside and outside of a picture and acts on the global mean. When the borders are faint, it works well, but when the boundaries are intense or no uniform, which are prevalent in practically all medical images, it fails miserably. Angle detection, rather than edge detection, is performed using a sobel operator in Edge Orientation Histogram (EOH) principles. It builds a histogram of the gradients of the edges by employing five distinct directions, namely horizontal, vertical, non-directional, and two diagonals [13]. The SIFT descriptor describes many object characteristics that are unaffected by various transformations.

Jinyu Wang et al. [15] investigated the various types of sketch illustration. For the feature extraction representation of drawings, the stroke fundamental unit of processing is used. Shape Feature Based, Stroke Description Based, and Combinatorial Primitives Based are the three approaches used to define image properties. However, both size and position are constrained. Sketch2image, a salient region-based segmentation algorithm based on keyword annotation and detection, was introduced by T. Chen [16]. Eitz et al. [17] present Photosketcher as a text-free interactive sketch synthesis system. Arbelaez et al. introduced another contour extraction approach termed gPb in [18], which took into account brightness texture and colour fluctuations in computations and attempted to solve some of the flaws of the canny edge recognition algorithm. GF-HoG in [19], the addition of spatial material to BoVW (Bag of Visual Word) in terms of an image descriptor. The application of neural networks to the image presented in [20] divides the image region-wise and predicts the bounding boundaries of respective areas with some probability, which function as weighted probabilities for such bounding boxes. Shape feature-based representation in [21] extracts local and global properties from any sketch to describe it. SaliencyCut adds spatial connectivity of image pixels based on the proposed area contrast histogram [22], which is used to calculate the saliency zone of pictures and sub images[23-25]. The well-known approach known as Lossless Data embedding [28] is used to eliminate distortion or data embedding in a tagged picture. It supports packaged, uncompressed, and transform formats like as PNG, BMP, and JPEG. The redundancy strategy is used to achieve a low distortion rate, with the difference of embedding expansion [29] focused on establishing a big embedding capacity. Reversibility is used to restore the original material. Vision-based masking [30] and quantization [31] are shown by Luminance Channel-based Algorithms. The density of messages is used to determine resilience in vision-based masking mode. In contrast to final bit embedding, the quantization method employs a single luminance value. In a Chrominance-channel-based approach [32], a watermark is inserted in the chrominance channel to create high-quality watermarked video material, and the same key is used for extraction. In the blue channel-based technique [33], the modulation is represented as a change in the pixel value of the blue channel. A Gaussian mask is used to compute equalisation in terms of Luminance intensity. The 3-color channel-based technique demonstrates coefficient quantization [34] and image splitting [35]. Watermarking on colour images is done by separating an image into well-separated patches and then calculating their decomposition to hide markings with no distortion within 3-colour channels. Reversible Data Hiding for Color Images is based on genetic algorithms for bitmaps [36], a prediction error system [37], and payload partitioning [38]. To increase inter-channel correlation payload partitioning is used, bitmaps are utilised to identify a bitmap for block compression, prediction error systems that incorporate separate data embedding techniques in each channel. SURF [39] is a scale and rotation invariant descriptor that measures repeatability, robustness, and uniqueness. It presents a replacement process for colour to accomplish a grey conversion by employing a chrominance value termed a single decomposition value in [40, 41]. Different Data encryption techniques [42] and Data Mining methods [43, 44] are studied.

## 3. Proposed Image and Data Retrieval

Proposed system is a hybrid module consisting of Image and Data retrieval through sketches. This methodology works efficiently with use of different methods like preprocessing, edge extraction, descriptor generation, constraint satisfaction as discussed below.

### 3.1. Sketches and Images

Query through sketch is an emerging area in the image retrieval world. The evolution started from Text-based query, then moved towards Content-based query, now extended with a Sketch based query system. Sketches are monochromatic images that show rough shapes with meaningful information that can be effectively used to retrieve real and natural images from a huge database. Preprocessing, edge extraction methods are applicable to sketches as well. Real images stored in a database are feature extracted images, thus called a feature database. Feature Database generation for proposed work is shown in Fig. 2. This database is further used in comparison of input sketches and images.

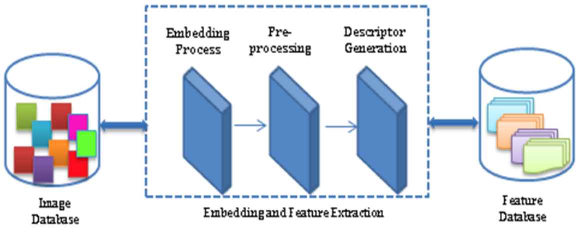



**Figure 2**. Feature database generation

### 3.2. Division Algorithm

This method separates the picture into two regions: areas where data can be inserted and areas where data cannot be inserted. According to the colour intensity values, the insertable region is again split into smooth and complex parts. Region with low intensity value is considered a smooth region, and further used for data embedding as smooth regions have less intensity variations. For calculating intensity differences, the central pixel is compared with pixels on Left, Right, Upper and Below as shown in Fig. 3. This difference is further compared with the set threshold to calculate smooth and complex regions in respective images. Equation (1) depicts the calculation of smooth and complex regions.

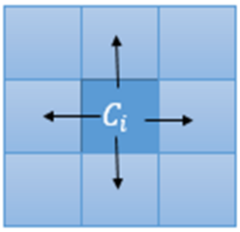



**Figure 3.** Reference pixel comparison

$$reference\ pixel = \begin{cases} Smooth & if\ difference < Threshold \\ Complex & Otherwise \end{cases} \quad (1)$$

where, *difference* is a summation of differences between the current reference pixel and the remote central pixel

### 3.3. Embedding Process

Red, Green, and Blue channels each contain in-color image data, and the sketch image key is encrypted at the receiver end to extract the data. A critical component in comparing a sketch and a real image is the grayscale representation of the image. Grayscale values should thus be preserved throughout the embedding and extracting processes. The extraction procedure decrypts encrypted communications; it receives encrypted photos as input and gives outputs as images with the extracted data.

In reference with steps explained [26], apply division algorithm on color image to extract R, G and B channels with smooth and complex regions for further embedding process. Each channel has 28 i.e. 256 bits. In the grayscale version of the image all three channels R, G and B have nearly the same color values. Embedding is done in all color channels after grayscale conversion. Least Significant Bit (LSB) is used for embedding here as shown in Fig. 4, as this does not affect much image color variation.


**Figure 4.** LSB bit used for embedding a character

Above figure shows Binary representation of color value as 10 and highlighted LSB bit has value 0. After embedding, if it's changed to 1 then the color value becomes 11. So there can be 3 possibilities like color value can be the same or it can decrement by one or it can be increased by one. By setting a threshold we can do embedding as per (2) and (3) in smooth and complex regions respectively.

$$Decimal\ (8\ bits) <= Threshold \quad (2)$$
$$Decimal\ (8\ bits) > Threshold \quad (3)$$

In 3 X 3 pixel, the first 8 bits of pixel are used to store Message Length (Li). Key (Ki) is stored starting from the 9th bit onwards with no specified limit, extended with actual Message(Mi) that can be stored upto last 256 bit. Separator '|' is used at the end of the key to distinguish between the key and message bits. Hence payload is calculated as shown in (4),

$$Payload = n * (|L_i| + |K_i| + |M_i|) \quad (4)$$

### 3.4. Pre-Processing

To reduce the amount of data in probed images without affecting their structural characteristics for additional processing, Canny Edge detector is utilised. It processed through various stages like smoothing of an image, gradient calculation w.r.t x and y, non-maxima suppression, two thresholding method and edge tracking.

Step 1: Blurring of the image to remove noise by applying a Gaussian filter with standard deviation σ, the image is first smoothed as shown in Fig. 5.

**Figure 5.** Smoothing of an image

Step 2: In an image having large gradient magnitudes are marked with strong edges with respect to both axes. Gradient magnitude is calculated by Euclidean distance as shown in (5), where $g_x$ and $g_y$ are gradients with respect to x and y. Direction of edges is also measured by angle theta and calculated as shown in (6).

$$Gradient\ Magnitude = |g_x| + |g_y| \quad (5)$$

$$Angle\ (\theta) = tan^{-1}\left(\frac{|g_y|}{|g_x|}\right) \quad (6)$$

Following Figure 6 shows a gradient image with respect to x and y.

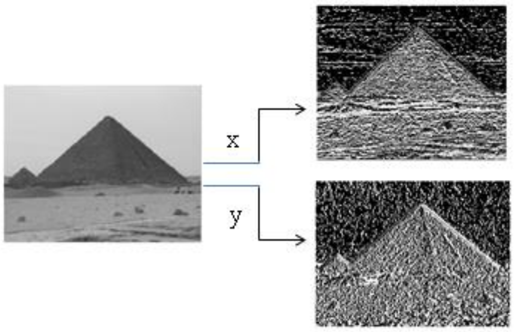
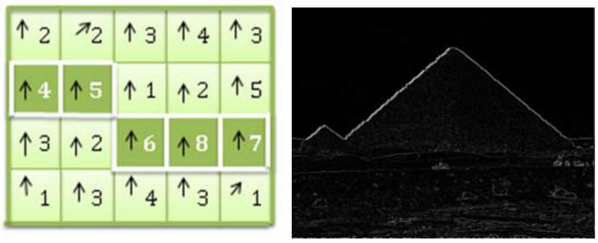



**Figure 6.** Gradient-Magnitude Image

Step 3: Make the gradient magnitudes "blurred" edges into "crisp" edges in the image. The edge strength of the current pixel should be compared to the edge strength of pixels in the positive and negative gradient directions as shown in Fig. 7(a). Preserve the value of the edge strength if the current pixel has the highest edge strength. If not, suppress the value. Fig. 7(b) shows sharp edges detected in sample image.




(a) (b)

**Figure 7.** Finding Edge Strength

Step 4: The strength of the pixel is marked in step 3, filtered to find strong shaping edges by applying two thresholds. Edge pixels larger than high thresholds are selected as strong whereas smaller than low thresholds are directly suppressed. Marking of edge pixels are shown in (7) where HT and LT represents High Threshold and Low Threshold respectively.

$$Edge\_Pixel = \begin{cases} Strong & if\ edge\ pixel \geq HT \\ Supressed & if\ edge\ pixel < LT \\ Weak & Otherwise \end{cases} \quad (7)$$

Step 5: Final edges are determined by suppressing all edges that are not connected to a very certain (strong) edge. This edge tracking is shown in Fig. 8.

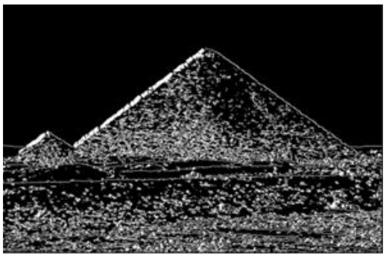


**Figure 8**. Edge tracking

Fig.9 is shows here canny edge detection of a sample input image after processing through the above five steps

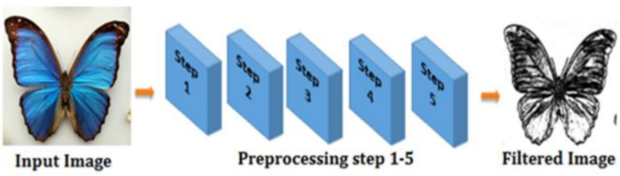



**Figure 9.** Filtered image after applying steps 1-5.

## 3.5 Descriptor generation

Histogram of Line Relationship (HLR) descriptor defined in [1], is used to capture the relationship among the line segments after the preprocessing of the image. Visual features of such lines are represented using a feature descriptor where an 8 block structure is used to capture the robust line relationship to extract the strong edges connecting to the central deployed line. This captured relationship is represented in a stream of numbers to distinguish it from others. As shown in Fig.10, blocks cover the left, right, upper and lower area and next four blocks are used to catch the lines which are close to the endpoints of the central line and cover the block borders.

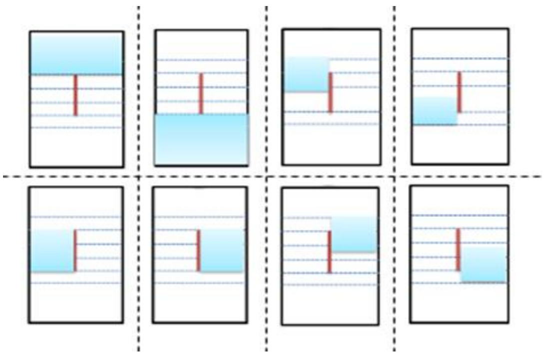


**Figure 10**. Descriptor covering 8 block boundaries

Descriptor generation algorithm is given and the Fig. 11 shows the representation of the number stream of the generated descriptor is shown and white block shows a line segment in an 8X8 block which is used for edge mapping.

---

Algorithm: Descriptor Generation

---

```
Get Edge details from Canny Edge Detection
Edges={Pre-processed Edges}
for each edge in edges
{
    Check if edge block Bi is not visited
    {
        Check for surrounding pixels
        Add edge block Bi to Descriptor
        Descriptor += {edge point with surrounding 8X8 pixel}
        Mark that edge block Bi as visited
    }
}
```

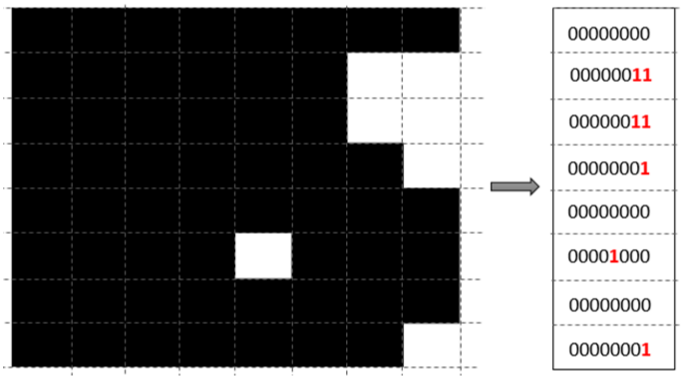



**Figure 11.** Bitwise Descriptor Generation Example

## 3.6 Constraint Satisfaction

To select object boundaries, noisy edges should be removed by removal process. The elimination of noisy edges immediately leads to the selection of real edges. True edges should be forecasted using spatial or coherent restrictions to eliminate false matches. Here, instead of spatial the coherent constraints are used with proposed edge length based constraint where length is compared to its orientations and edges are picked as true matching if they meet the threshold value**.**

### 3.6.1 Coherent Constraint

In this work, edges are described by their line relationship in terms of a series of line segments; such a coherent relationship is defined in [1]. If two descriptor words appear on one edge of a sketch image, the corresponding descriptor words appear on one edge of the Real image. This coherency is calculated with constraint as per (8),

$$CoherentValue = \begin{cases} 1 & if\ loc(Es) = loc(Ep) \\ 0 & Otherwise \end{cases} \quad (8)$$

Where Es represents words on one edge of Sketch image and Ep represents words on same edge of Real image.

### 3.6.2 Edge Length Constraint

The coherent constraint alone is not enough to filter the false matches as here transformation and scaling is not considered. Proposed edge length constraint works on location, direction and length of an edge by comparing with threshold it removes false matches. Equation (9) represents the count match of the constraint with respect to the set threshold and countMatch is calculated as,

$$\sum_{edge}^{SS}\sum_{edge1}^{RS}(countMatch = countMatch + 1)$$
$$\begin{cases} if\ |loc(edge) - loc(edge1)| < Threshold\ \wedge \\ if\ |direction(edge) - direction(edge1)| < Threshold\ \wedge \\ if\ |len(edge) - len(edge1)| < Threshold \end{cases} \quad (9)$$

Where, SS is total edges in the sketch image and RS is total edges in the Real image.

### 3.7 Extraction Process

Extraction process is applied on encrypted image I'. Division algorithm is applied on encrypted image I'. Key is extracted through LSB bit and divides insertable area into smooth and complex regions. By considering non centred pixel compute the difference value till the key get retrieved. Message can get extracted from all three channels of real image. Mathematical formulation of extraction process is shown in [26].

## 4. Experiments and Results

### 4.1 Dataset

Three datasets can be used for performance evaluation of the proposed work: Hu[ ] given one dataset, Flicker 15K, are publicly available. Two more datasets, Caltech256 and Flicker 3M, generated by Wang.et.al.[1] can be used for evaluation here. As per requirement, the same dataset should be used for this hybrid model which is the combination of Data and Image retrieval. Therefore, Flicker 15K is found as a suitable dataset for this hybrid model. Dataset Flicker 15K contains 33 abject categories and approximately 10 sketches for each category. This dataset extended to 330 sketches as required as input images. Fig.12 shows sample sketches from a used dataset.

**Figure 12.** Sample input sketches

### 4.2 Implementation with Thresholds

a) Pre-processing: It is carried out with the help of a canny edge detector here. Gaussian mask and sigma value used is 30 and 50 respectively. Two thresholds are required for edge tracking. These two thresholds i.e. Low and High thresholds are set to 10 and 30 pixels respectively.
b) Descriptor: The descriptor is designed to capture the line relationship, therefore considering the scaling factor we use the 64 pixels as a threshold to restrict the maximal length of line segments.
c) Edge Length constraint: Edge Length based constraint is set to threshold value as 5 super pixels in proposed work.

### 4.3 Parameters

a) Descriptor size: The descriptor size is calculated by the size of the blocks, which is determined here as 8x8 blocks.
b) Message Length: This parameter is used in the embedding process and also used in calculating the payload bits. According to message length the payload bits can vary. As in this framework we are using 8 bit representation of RGB color channels. Therefore, a maximum of 256 characters can be stored in each channel. But while maintaining the grayscale invariance, here RGB values are the same. Hence 0-255 characters can be stored in the embedding process including message length value and message key.
c) Feature Extraction: It is calculated by the average feature extraction time required by one image. The fair comparison of feature extraction cannot be given as descriptors are generated after message embedding. The average feature extraction time of one image with HLR descriptors after embedding is 0.38 seconds, which is still less if compared with GF-HoG without embedding, which takes 0.4 seconds.
d) Memory Usage: Memory use is determined by the number of descriptors created and the amount of additional details saved. On an average, 2000 descriptors are generated for a single real image and 500 descriptors for sketch images in this framework.
e) MSE: It is a measure of the quality and performance of an estimator. Here, it shows the mean of the square of difference between actual pixel value and estimated value and calculated as given in (10),

$$MSE = \frac{1}{M\ X\ N}\sum_{i=1}^{M}\sum_{j=1}^{N}(I_{ij} - I'_{ij})^2 \quad (10)$$

where, M X N is size of image, I is pixel value of original cover image and I' is pixel value of stego image

f) Payload calculation: payload is calculated as shown in equation (4), where, n is no of bits. If message of 196 characters (including key, message length and message itself) is to be stored and each character is stored with 8 bits then payload is calculated as 196*8=1568 bits. Sample images from the data set with message length and its payload bits are shown in Table I.

**Table I.** Sample data set categories with payload bits

| Sr. No. | Data set Category | Message Length | Payload Bits |
|---|---|---|---|
| 1 | Architecture 19 | 136 | 1088 |
| 2 | Butterfly | 123 | 984 |
| 3 | big_ben | 131 | 1048 |
| 4 | Sunflower | 161 | 1288 |
| 5 | Indian_arch | 88 | 704 |

### 4.4 Evaluation Metrics

**4.4.1 PSNR:** Peak Signal to Noise Ratio (PSNR) is calculated as shown in (11), where Max is calculated as $2^{n-1}$ here n=8. Therefore, Max is 255 for proposed work. Fig. 13 shows Payload vs PSNR representation for data set image categories.

$$PSNR = 10 * log_{10}\ (Max^2 / MSE) \quad (11)$$

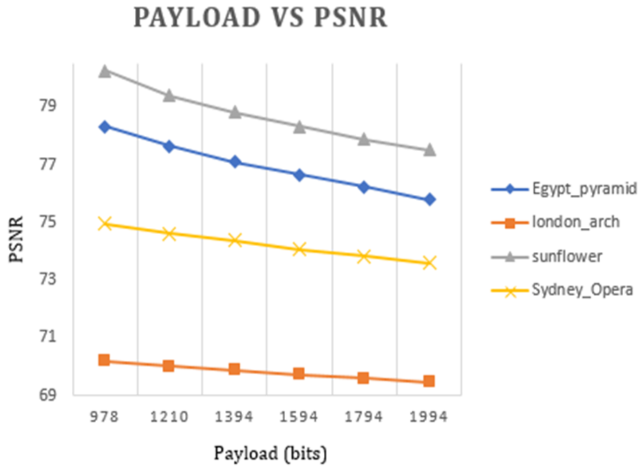



**Figure 13.** Payload Vs PSNR performance on grayscale images

PSNR values after vacating rooms and after recovery for sample dataset categories are shown in Fig. 14.

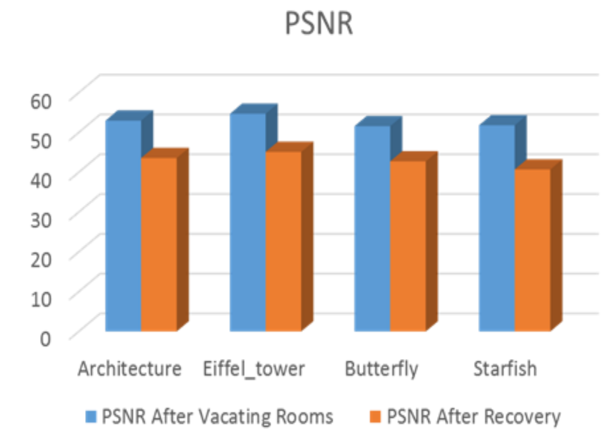



**Figure 14.** PSNR calculated after Vacating rooms and after Recovery

4.4.2 Precision and Recall**:** Widely accepted evaluation metrics called Precision and Recall are used here for performance evaluation of image retrieval systems. Table II shows Precision and Recall calculation for sample images categories. Precision vs Recall curves are also plotted as shown in Fig. 15. Precision and recall of the retrieval system is calculated as follows,

**Precision** = (Number of relevant and retrieved images) / Number of total retrieved images

**Recall** = (Number of relevant and retrieved images) / Number of possible relevant images.

**Table II**. Precision and Recall calculation for sample dataset categories

| Sr. No. | Data set Category | Retrieved Images | Relevant Images | Relevant Retrieved Images | Precision | Recall |
|---|---|---|---|---|---|---|
| 1 | fire_balloon | 10 | 12 | 9 | 0.90 | 0.75 |
| 2 | Egypt_pyramid | 11 | 13 | 9 | 0.82 | 0.69 |
| 3 | notre_dame_paris | 13 | 14 | 10 | 0.77 | 0.71 |
| 4 | eiffel_tower | 10 | 9 | 8 | 0.80 | 0.89 |
| 5 | airplane | 10 | 12 | 9 | 0.90 | 0.75 |

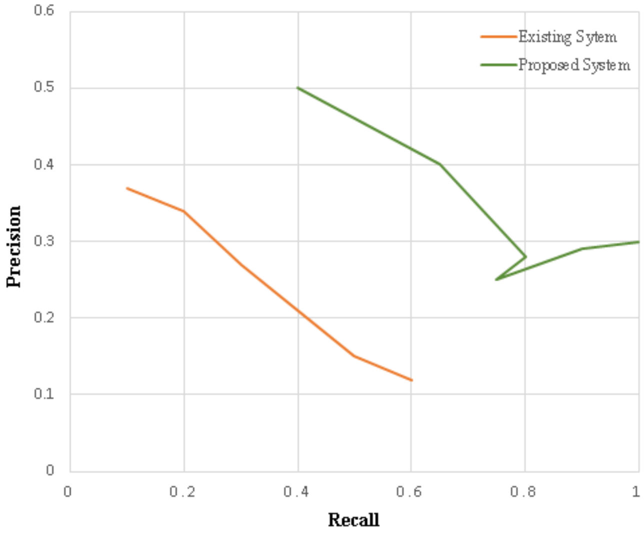



**Figure 15.** Precision Vs Recall curve

4.4.3 Retrieval Time and Retrieved Images

Average retrieval time required for image and data retrieval is 90 seconds. Input sketch and output images of proposed retrieval is given in Fig. 16.

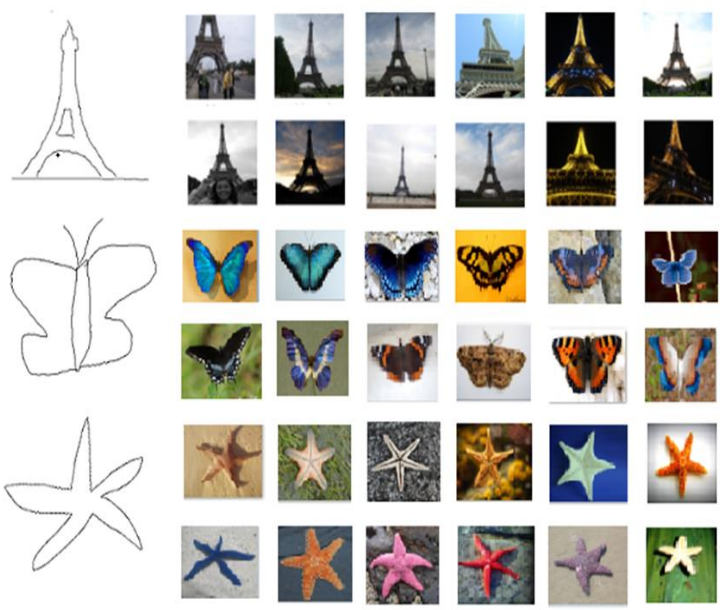


**Figure 16.** Retrieved matches for given sketch image

## 5. Conclusion

The proposed hybrid approach treats image retrieval with new angles by extending it with data retrieval with grayscale invariance throughout the process. HLR Descriptor used to capture the line relationships. Proposed edge length-based constraints are applied with coherent parts to capture high retrieval performance by filtering the false matches. To minimise the data loss during data hiding is provided by applying the Reversible Data Hiding technique through the grey version of the image. This combination named as "Sketch-Based Image and Data Retrieval (SBIDR)" works efficiently and in terms of data and image extraction. The experimental results validate the efficiency of the proposed framework.

## References


[1] Shu Wang, Jian Zhang, Tony X. Han and Zhenjiang Miao, "Sketch Based Image Retrieval Through Hypothesis-Driven Object Boundary Selection With HLR Descriptor" , IEEE transactions on Multimedia, Vol. 17, No. 7 July 2015.

[2] Dongdong Hou, Weiming Zhang, Kejiang Chen, Sian-Jheng Lin and Nenghai Yu, " Reversible Data Hiding in Color Image with Grayscale Invariance." , in Transaction on circuits and Systems for Video Technology. 1051-8215, 2018.

[3] Kede Ma, Weiming Zhang, Xianfeng Zhao, Nenghai Yu, and Fenghua Li, "Reversible Data Hiding in Encrypted Images by Reserving Room Before Encryption", IEEE Transactions on Information Forensics and Security, Vol. 8, No. 3, March 2013.

[4] J. Canny, ''A computational approach to edge detection,'' IEEE Trans. Pattern Anal. Mach. Intell., vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.

[5] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa, "A descriptor for large scale image retrieval based on sketched feature lines,",in Proc. 6th Eurograph. Symp. Sketch-Based Interfaces Modeling, 2009, pp. 2936.

[6] R. Hu, M. Barnard, and J. Collomosse, ''Gradient field descriptor for sketch based retrieval and localization,'' in Proc. IEEE Int. Conf. Image Process. (ICIP), Sep. 2010, pp. 1025–1028.

[7] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa, ''Sketch-based image retrieval: Benchmark and bag-of-features descriptors,'' IEEE Trans. Vis. Comput. Graphics, vol. 17, no. 11, pp. 1624–1636, Nov. 2011.

[8] Y. Cao, C. Wang, L. Zhang and L.Zhang, "Edgel index for large – scale Sketch-bassed image search," in Proc. IEEE conf. omput. Vis Pattern Recognit.(CVPR), Jun. 2011, pp. 761-768.

[9] K. Bozas and E. Izquierdo, "Large scale sketch based image retrieval using patch hashing," in Adv. Visual Comput., vol. 7431, pp. 210219, 2012.

[10] R. Hu and J. Collomosse, "A performance evaluation of gradient field hog descriptor for sketch based image retrieval." in Comput. Vis. Image Understand., vol. 117, no. 7, pp. 790806, 2013.

[11] C. Yang, O. Tiebe, P. Pietsch, C. Feinen, U. Kelter, and M. Grzegorzek,''Shape-based object retrieval by contour segment matching,'' in Proc. IEEE Int. Conf. Image Process. (ICIP), Oct. 2014, pp. 2202–2206.

[12] Tingting Liu, Haiyong Xu,"Medical Image Segmentation Based on a Hybrid Region-Based Active Contour Model", Research Article, Comput. Math Methods in Med., 2014.

[13] Y H Sharath Kumara, D S Gurub, "Retrieval of Flower Based on Sketches," in International Conference on Information and Communication Technologies, Procedia Computer Science 46 (2015 ) 1577 – 1584.

[14] C. Xiao, C. Wang, L. Zhang, and L. Zhang, ''Sketch-based image retrieval via shape words,'' in Proc. ACM Int. Conf. Multimedia Retr. (ICMR), 2015,pp. 571–574.

[15] Jingyu Wang, Yu Zhao , Qi Qi, Qiming Huo, Jian Zou, Ce Ge, And Jianxin Liao, "MindCamera: Interactive Sketch-Based Image Retrieval and Synthesis." , in Special Section On Recent Advantages Of Computer Vision Based On Chinese Conference On Computer Vision (CCCV) 2017, vol 6, 2018.

[16] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, and S.-M. Hu, ``Sketch2Photo: Internet image montage," ACM Trans. Graph., vol. 28, no. 5, Dec. 2009,Art. no. 124.

[17] M. Eitz, R. Richter, K. Hildebrand, T. Boubekeur, and M. Alexa,``Photosketcher: Interactive sketch-based image synthesis," IEEEComput. Graph. Appl., vol. 31, no. 6, pp. 56-66, Nov./Dec. 2011.

[18] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, ``Contour detection and hierarchical image segmentation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 5, pp. 898-916, May 2011.

[19] T. Bui and J. Collomosse, ``Scalable sketch-based image retrieval using color gradient features," in Proc. IEEE Int. Conf. Comput. Vis. (ICCV)Workshop, Dec. 2015, pp. 1012-1019.

[20] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, ``You only look once:Unied, real-time object detection," in Proc. IEEE Conf. Comput. Vis.Pattern Recognit. (CVPR), Jun. 2016, pp. 779-788.

[21] P. Xu et al., ``Cross-modal subspace learning for ne-grained sketch-based image retrieval," Neurocomputing, vol. 278, pp. 7586, Feb. 2018, doi: 10.1016/j.neucom.2017.05.099.

[22] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S.-M. Hu, ``Global contrast based salient region detection," IEEE Trans. Pattern Anal. Mach. Intell., vol. 37, no. 3, pp. 569582, Mar. 2015.

[23] S. Parui and A. Mittal, "Similarity-invariant sketch-based image retrieval in large databases," in Proc. 13th Eur. Conf. Comput. Vis. Conf. Comput. Vis., 2014, vol. 8694, pp. 398–414.[31]

[24] C. Ma, X. Yang, C. Zhang, X. Ruan, M.-H. Yang, and O. Coporation, "Sketch retrieval via dense stroke features," in Proc. Brit. Mach. Vis.Conf., 2013, vol. 2, pp. 65.1–65.11. [9]

[25] X. Cao, H. Zhang, S. Liu, X. Guo, and L. Lin, "Sym-fish: A symmetry aware flip invariant sketch histogram shape descriptor," in Proc. IEEE Int. Conf. Comput. Vis., Dec. 2013, pp. 313–320.

[26] D. Birari, D. Hiran, V. Narawade, "Sketch Based Data Retrieval using Reversible Data Hiding on Images", in JUSST vol 22, ISSN 1007- 6735, Dec 2020.

[27] D. Birari, D. Hiran, V. Narawade "Image Retrieval through sketches based on Descriptor with Data Retrieval using Reversibility Method" in Proc. IEEE Int. Conf. for Advancement in Technology, Jan 2022.

[28] J. Fridrich and M. Goljan, "Lossless Data Embedding for All Image Formats," in SPIE Proceedings of Photonics West, Electronic Imaging, Security and Watermarking of Multimedia Contents, vol. 4675, pp. 572- 583, San Jose, Jan. 2002.

[29] J. Tian, "Reversible Data Embedding Using a Difference Expansion," IEEE Trans. Circuits System and Video Technology, vol. 13, no. 8, pp. 890-896, Aug. 2003.

[30] M. Kutter and S. Winkler, "A vision-based masking model for spread- spectrum image watermarking," IEEE Trans. Image Processing, vol. 11, no. 1, pp. 16-25, Jan. 2002.

[31] P. Bao and X. Ma, "Image adaptive watermarking using wavelet domain singular value decomposition," IEEE Trans. Circuits and Systems for Video Technology, vol. 15, no. 1, pp. 96-102, Jan. 2005.

[32] M. Asikuzzaman, M. J. Alam, A. J. Lambert, and M. R. Pickering, "Imperceptible and robust blind video watermarking using chrominance embedding: A set of approaches in the DT CWT domain," IEEE Trans. Information Forensics and Security, vol. 9, no. 9, pp. 1502-1517, Sep. 2014

[33] M. R. A.Lari, S. Ghofrani, and D. McLernon, "Using curvelet transform for watermarking based on amplitude modulation," Signal, Image and Video Processing, vol. 8, no. 4, pp. 687-697, May 2014.

[34] C. H. Chou and K. C. Liu, "A perceptually tuned watermarking scheme for color images," IEEE Trans. Image Processing, vol. 19, no. 11, pp. 2966-2982, Nov. 2010.

[35] Y. He, W. Liang, J. Liang, and M. Pei, "Tensor decomposition based color image watermarking," Proceedings of SPIE, vol. 9069, pp. 90690U- 90690U-6, Jan. 2014.

[36] C. Chang, C. Lin, Y. Fan, "Lossless data hiding for color images

based on block truncation coding," Pattern Recognition, vol. 41, no. 7, pp. 2347-2357, 2008.

[37] J. Li, X. Li, and B. Yang, "Reversible data hiding scheme for color image based on prediction-error expansion and cross-channel correlation," Signal Processing, vol. 93, no. 9, pp. 2748-2758, 2013.

[38] B. Ou, X. Li, Y. Zhao, and R. Ni, "Efficient color image reversible data hiding based on channel-dependent payload partition and adaptive embedding," Signal Processing, vol. 108, pp. 642-657, 2015.

[39] H. Bay, T. Tuytelaars, and L. J. V. Gool, SURF: Speeded up robust features, in Proc. Eur. Conf. Comput. Vis., pp. 404-417, 2008.

[40] Sowmya V, Govind D, Soman K P, "Significance of incorporating chrominance information for effective color-to-grayscale image conversion," Signal, Image and Video Processing, vol. 11, pp. 129-136, 2017.

[41] D. Birari, D. Hiran, V. Narawade, "A Survey on Sketch Based Image and Data Retrieval." In Proc. Springer Int. Conf. on Communication and Cyber Physical Engg. (ICCCE), vol 570, pp. 285-290, 2019.

[42] D. P. Gadekar, S N Popat, A. H. Raut, "Exploring Data Security Scheme into Cloud Using Encryption Algorithms" International Journal of Recent Technology and Engineering , Volume-8 Issue-2, July2019,

[43] S N Popat, Y. P. Singh," Efficient Research on the Relationship Standard Mining Calculations in Data Mining" in Journal of Advances in Science and Technology | Science & Technology, Vol. 14, Issue No. 2, September-2017,

[44] S N Popat*, Y. P. Singh," Analysis and Study on the Classifier Based Data Mining Methods" in Journal of Advances in Science and Technology Vol. 14, Issue No. 2, September-2017.