International Journal of Innovative Research in Computer Science and Technology (IJIRCST)
ISSN(Online): 2347-5552, Volume-12, Issue-3, May 2024
https://doi.org/10.55524/ijircst.2024.12.3.3
Article ID IRP1444, Pages 11-16
www.ijircst.org

Design And Development of Sketch Based Image Retrieval Using Deep Learning

Dr S. A. Talekar[1], Shravani A. Lajurkar[2], Divya S. Patil[3], Rutika A. Benke[4], and Pranjal A. Kunde[5]

[1,2,3,4,5] Department of Information Technology, Savitribai Phule Pune University(SPPU), Pune, Maharashtra India

Correspondence should be addressed to Dr. S A Talekar; talekar.sopan@kbtcoe.org

Received 15 March 2024; Revised 28 March 2024; Accepted 8 April 2024

ABSTRACT—In this cutting edge, the common wrong doing rate is expanding day-by-day and to manage up with this the criminal divisions as well ought to discover ways in which would speed up the by and large preparation and offer assistance in bringing one to justice. In response to rising crime rates, law enforcement agencies are turning to advanced algorithms capable of matching freehand sketches with images in databases. These algorithms, utilizing sophisticated feature extraction techniques and deep learning models, significantly enhance identification accuracy. By leveraging Sketch based image retrieval technology, investigations are expedited, leading to quicker suspect apprehensions and resolution of criminal cases. This results in improved public safety and justice outcomes, as well as more efficient law enforcement practices overall.

KEYWORDS— Forensic Face Sketch, Face Sketch Construction, Face Recognition, Criminal Identification, Deep Learning, Machine Locking.

I. INTRODUCTION

A developing number of criminal exercises are taking place each day [1]. Hoodlums are being followed down and captured by law requirement specialists. The specialist located an assortment of clues at the wrongdoing scene to recognize the criminal, counting an eyewitness, Closed-Circuit TV (CCTV) recordings, Deoxyribonucleic corrosive (DNA) samples, etc. Observers are among the foremost predominant sources of proof. Numerous applications are utilizing facial acknowledgment frameworks [2]. To distinguish the suspect utilizing an onlooker as well as facial acknowledgment innovation, an outline of the suspect was necessary. Sketch-based picture recovery (SBIR) has earned noteworthy consideration in later a long time as an inventive approach to bridge the crevice between human draws and computerized images, revolutionizing the way clients are associated with picture databases[3]. With the proliferation of touch-based gadgets and stylus-equipped interfacing, the request for productive systems able of recovering pictures based on hand-drawn draws has heightens over various areas, counting craftsmanship, plan, and law requirement [4].

A criminal can be effectively distinguished and brought to equity employing a confront portray drawn based on the data been given by the eyewitness, be that as it may in this world of modernization the traditional way of hand drawing a portrayal isn't found to be that successful and time sparing when utilized for coordinating and distinguishing from the as of now accessible database or real-time databases [5]. Amid the past there were a few procedures proposed to change hand-drawn confrontation outlines and utilize them to naturally recognize and recognize the suspect from the police database, but these techniques might not give the wanted exactness[6].

A. Objectives

- Facilitate Investigations
- Crime Prevention
- Continuous Improvement
- Enhanced Composite Sketches

II. RELATED WORK

Through the literature survey various insights are being found while implementing this project. Following are the main insights that are found through different papers:

Current criminal sketches rely on eyewitness memories, resulting in inaccuracies[7]. The system for retrieving evidence photos needs improvement [17]. Existing studies mostly focus on facial features, neAglecting additional information like skin and eye color [8]. Enhancing the inclusion of these details is crucial for more effective law enforcement.Khan et al. developed a method for retrieving mugshot images from sketch images using a Bayesian classification-based approach, focusing on the local aspects of mug-shots and sketches [9]. Additionally, in their work, they introduced a method to retrieve suspects' photos based on linguistic descriptions [10]. This involved converting linguistic input into facial attributes, generating a facial attribute vector, and employing a cluster-based ensemble classification approach to match the input description with relevant face photos from the database. Jain et al. developed SeekSuspect, an interactive suspect retrieval system relying solely on an informant's visual memory [11]. Sagayam et al. enhanced content-based image retrieval with a semantic approach, incorporating 3D characteristics for improved performance. Suwannakhun et al. proposed a system combining a geometric face model with an identification system to minimize retrieval errors by cross-referencing the person's identity on an ID card with the existing face image database [12].

Chuo et al. [14] introduced a model for suspicious face detection, monitoring and comparing suspect faces across multiple surveillance cameras to track suspicious activities.Shrivastava et al. proposed a face retrieval system [13]. The input in their approach consists of visual inputs from

users and attempts to retrieve the required target face image[16].The results of the experiments were based on a small dataset, which is one of the drawbacks of their approach. Ounachad et al. have proposed a sketch-to-face retrieval approach. Based on the sketch, it retrieves the suspect's face image from a database. A face image was retrieved using Euclidean distance, Murkowski distance,
Manhattan distance, and Chebyshev distance [15]. However, sketch-based face identification could be hindered by the noise provided by eyewitness reports. Avoid this noisy information by eyewitnesses and retrieve the most relevant suspect face image.The proposed work accepts verbal descriptions as inputs and finds the most appropriate image from the face repository.

III. DESIGN AND IMPLEMENTATION OF THE PROPOSED WORK

System Architecture represents the structural design of the system. The system architecture of the Sketch Based Image Retrieval is represented in Figure 1.

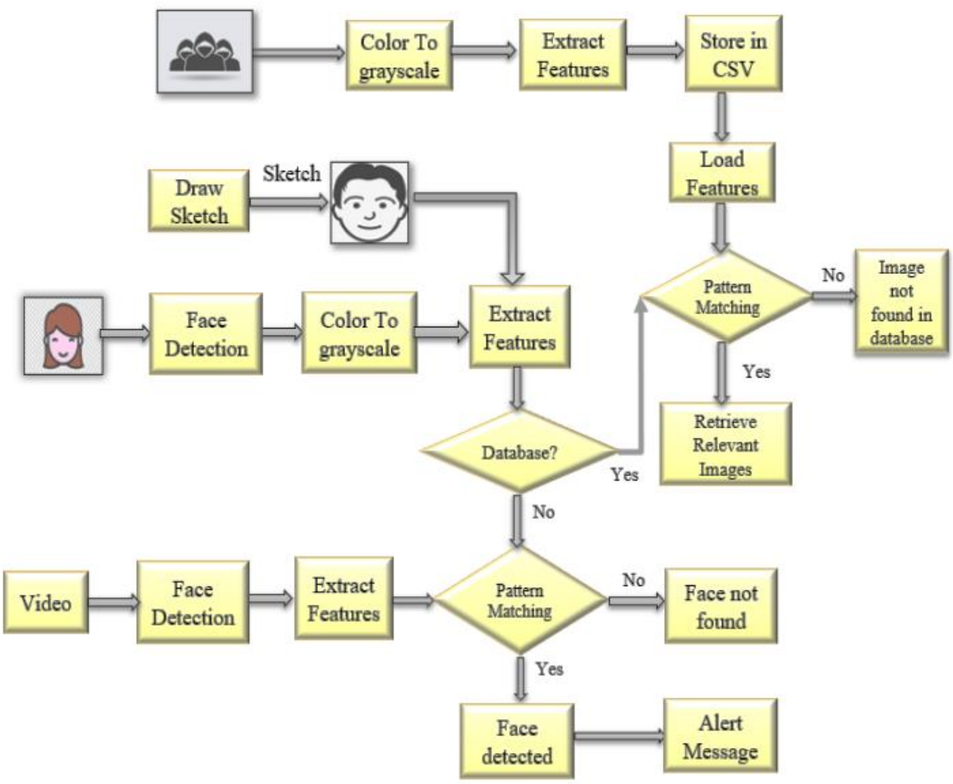

Figure 1: Architecture (SIBR)

As shown in figure (1.1) the system initiates by converting color images to a binary scale (0 to 1) through a specialized color-to-grayscale algorithm[18]. Resultant black and white images are subjected to feature extraction, and the discerned attributes are stored in a CSV file. The feature-rich image repository is seamlessly integrated into Firebase Cloud for efficient storage and retrieval.
Two input types, sketches and images, are accommodated. Sketches are generated via a drag-and-drop interface guided by victim descriptions. Feature extraction and pattern matching identify relevant images, providing a percentage-based match indication[19]. For input images, facial detection is employed. Images are transformed to the binary scale, and feature extraction, followed by pattern matching, retrieves pertinent images with a percentage-based match assessment. This system offers a comprehensive solution for Sketch-Based Image Retrieval with Firebase Cloud integration.

A. *Deep Learning Library used: Dlib*

Dlib is a powerful C# library that provides a wide range of Deep learning and computer vision algorithms. It's often used for tasks like facial landmark detection, object detection, image processing, and more.

Functionalities of Dlib used in SIBR:

- ***Dlib.detectface<RgbPixel> :*** The detectface function analyzes the image data (assuming it's in RGB format) using a deep learning model or other computer vision algorithms trained to identify faces.
- ***Dlib.LoadImage<RgbPixel>(faceImagePath):*** Dlib likely has internal mechanisms to load the image from a file path you provided earlier (not explicitly shown in this function call).
- ***Dlib_GetFacialLandmarks(imagePtr):*** Get Facial Landmarks is used for facial landmark detection in an image.

IV. PSEUDOCODE OF ALGORITHMS

Step 1: Color-To-Gray-Scale

Converting an image from color to grayscale involves transforming each pixel's color information into a single intensity value. Grayscale images represent shades of gray, with black and white being two extreme values, and various shades of gray in between, depending on the intensity. Pseudocode as shown in table 1.

Table 1: Color to grayscale

```
Load the color image
    Bitmap colorImage = new Bitmap("input.jpg");
    // Convert the color image to grayscale
    Bitmap grayscaleImage = ConvertToGrayscale(colorImage);
    // Save the resulting grayscale image
    grayscaleImage.Save("output_grayscale.jpg");
            static Bitmap ConvertToGrayscale(Bitmap
colorImage{
    Bitmap grayscaleImage = new Bitmap(colorImage.Width,
colorImage.Height);
    for (int x = 0; x < colorImage.Width; x++){
        for (int y = 0; y < colorImage.Height; y++) {
        Color pixelColor = colorImage.GetPixel(x, y);
        // Luminosity method: weighted sum of RGB values
        int grayscaleValue = (int)(0.299 * pixelColor.R + 0.587 *
pixelColor.G + 0.114 * pixelColor.B);
        Color grayscaleColor = Color.FromArgb(grayscaleValue,
grayscaleValue, grayscaleValue);
        grayscaleImage.SetPixel(x, y, grayscaleColor);
      }
    return grayscaleImage;
```

Explanation: This pseudocode uses the Bitmap class in C# for image processing. Adjustments might be necessary depending on your specific use case, and you might want to explore more advanced image processing libraries for more efficient solutions. If "SBIR" is a specific algorithm, please provide additional details for more accurate assistance.

Step 2: Feature Extraction

Feature extraction in the context of face images refers to the process of capturing and representing the distinctive characteristics or patterns from facial images that are relevant for a specific task, such as facial recognition, emotion analysis, or age estimation. These features are often used to create a compact and informative representation of the face, facilitating efficient and effective analysis.Pseudocode as shown in table 2.

Table 2: Feature Extraction

```
            static void Main()
            {
    // LoaaceImage =
Dlib.LoadImage<RgbPixel>(faceImagePath);
            var faceImagePath =
        "path/to/your/face_image.jpg";
            var f// Use a pre-trained shape predictor model
    var shapePredictorPath =
"path/to/shape_predictor_68_face_landmarks.dat";   var
shapePredictor =
ShapePredictor.Deserialize(shapePredictorPath);
            // Detect facial landmarks
            var faceLandmarks =
        shapePredictor.Detect(faceImage);
            // Extract features from facial landmarks
            var facialFeatures =
        ExtractFeatures(faceLandmarks);
            // Print or use the extracted features
            foreach (var feature in facialFeatures)
            {
              Console.WriteLine($"Feature: {feature}");
            }
        }
        static List<Point>
    ExtractFeatures(FullObjectDetection
    faceLandmarks)
        {
            // Extract features based on the facial landmarks
            var features = new List<Point>();
                return features;
        }
```

Step 3: Pattern matching

In the context of face recognition, pattern matching refers to the process of comparing the facial features or patterns extracted from an input face image with those stored in a database to determine the identity of the individual. The goal is to find a match or similarity measure that indicates how closely the input face aligns with the stored templates or representations. Pseudocode is shown in table 3.

Table 3: Pattern Matching

```
  // Match an input face against known faces
    public string MatchFace(List<double>
inputFaceFeatures){
        double bestMatchScore = double.MaxValue;
        string bestMatchPerson = null;
        foreach (var entry in knownFaces){
          string personName = entry.Key;
          List<double> knownFaceFeatures = entry.Value;
          double matchScore =
CalculateFeatureDifference(inputFaceFeatures,
knownFaceFeatures);
          if (matchScore < bestMatchScore){
            bestMatchScore = matchScore;
            bestMatchPerson = personName;
          }
        }
        return bestMatchPerson;
    }
          // Calculate the difference between two sets of face
        features
          private                                      double
        CalculateFeatureDifference(List<double>   features1,
        List<double> features2)
          {
```

```
            double sumSquaredDifferences = 0.0;

            for (int i = 0; i < features1.Count; i++)
            {
                double difference = features1[i] - features2[i];
                sumSquaredDifferences += difference * difference;
            }

    return Math.Sqrt(sumSquaredDifferences);
        }
    }
    class Program{
        static void Main(){
            // Create a FaceMatcher instance
            var faceMatcher = new FaceMatcher();
            // Add known faces with their features
            faceMatcher.AddKnownFace("PersonA", new List<double> { 0.1, 0.2, 0.3 });
          faceMatcher.AddKnownFace("PersonB", new
                List<double> { 0.4, 0.5, 0.6 });
            // Extract features from an input face (you would use a real feature extraction method here)
            var inputFaceFeatures = new List<double> { 0.2, 0.3, 0.4 };
            // Match the input face against known faces
            string matchedPerson = faceMatcher.MatchFace(inputFaceFeatures);
            // Print the result
            Console.WriteLine($"Matched person: {matchedPerson}");
        }
```

The Euclidean distance formula is commonly used for pattern matching and measuring the similarity or dissimilarity between two vectors or points in a multi-dimensional space. The formula is as follows:

Euclidean Distance:

$$d(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2} \quad \text{-------- } 1$$

Euclidean Distance: Formula Used in implementation is :

var diff = queryFeature - storeFeature ---------- (2)

V. RESULT AND DISCUSSION

Following Figures shows the results. Figure 2 shows the splash image code in which a new splash function is used to show the initial image for 3 sec.

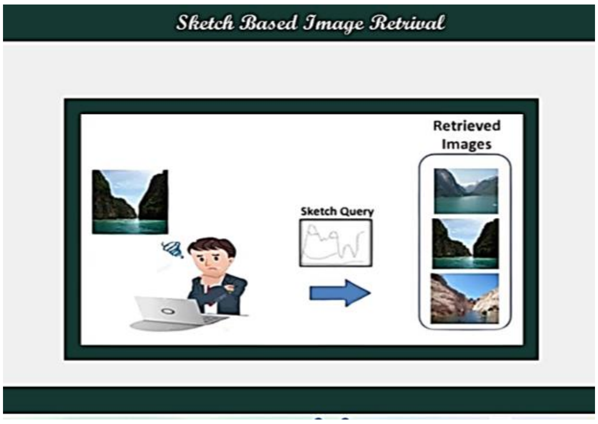

Figure 2: .Splash Image

Figure 3 shows the admin login where admin can login to the system

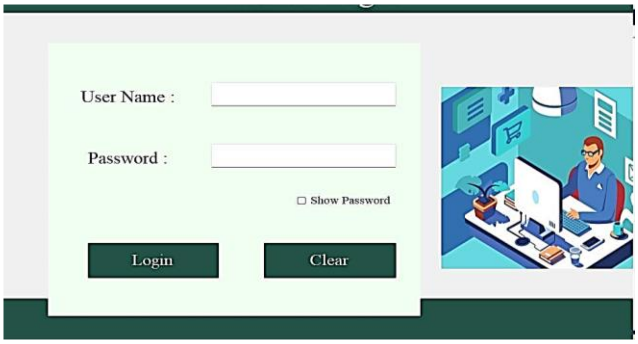

Figure 3: Login Page

Figure 4 shows the registration page where admin can register to the new criminal.

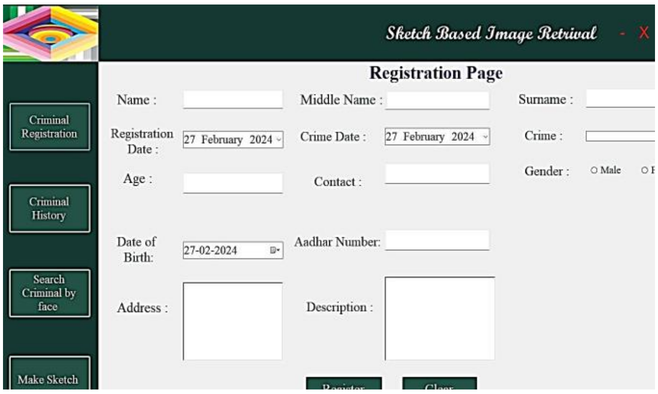

Figure 4: Registration Page

Figure 5 shows the face detection module in which the face of the criminal can be detected and stored in the database with the help of local storage and camera.

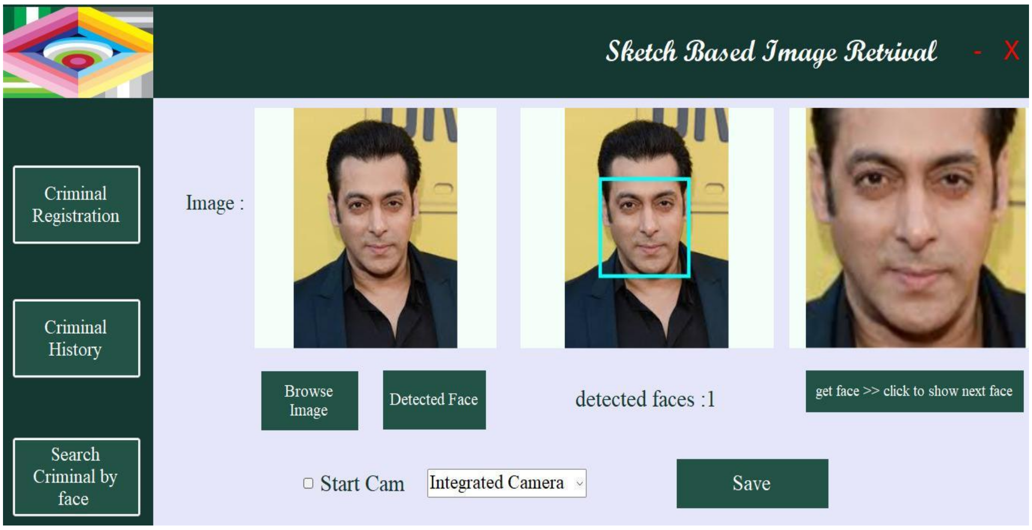

Figure 5: Face Detection

Figure 6 shows all criminals history in tabular format.

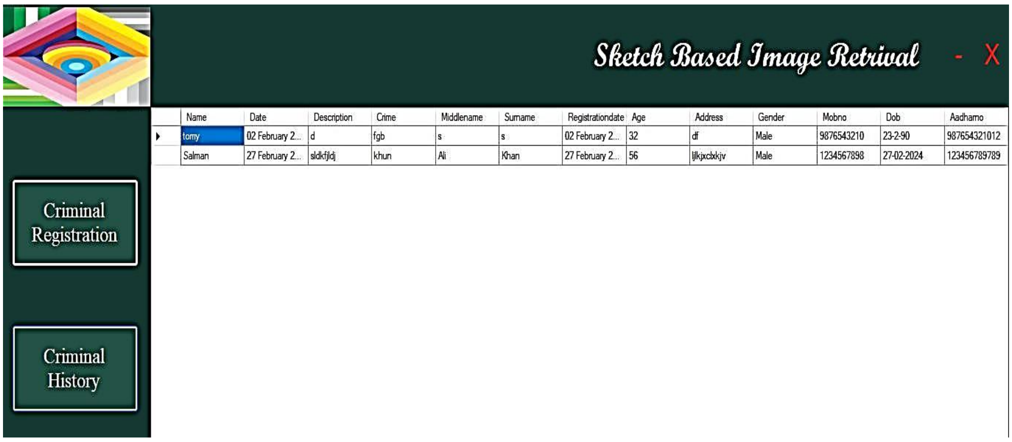

Figure 6:.Criminal History

Figure 7 shows face detection module where image and sketch is given as input then same or face with low dissimilarity retrieved as output from database with name and similarity percentage.

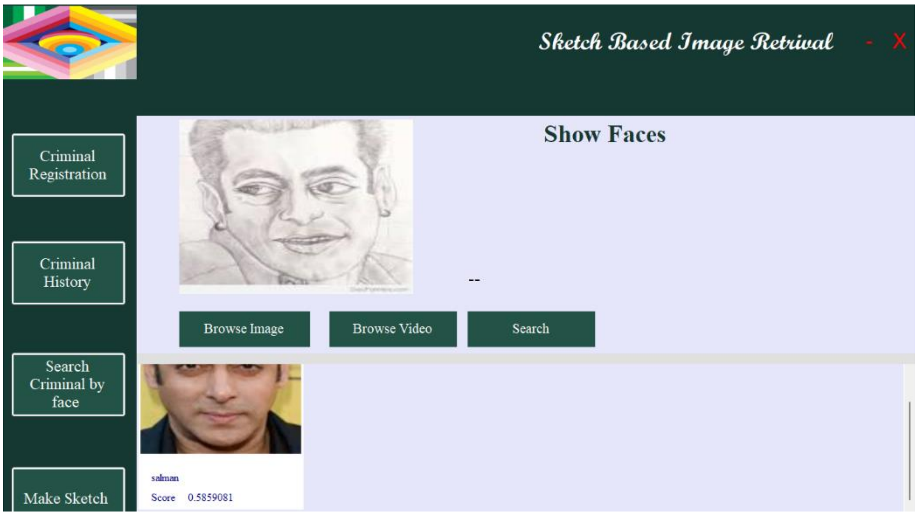

Figure 7: Face Detection

VI. CONCLUSION

The "Sketch Based Image Retrieval" project was designed, developed and finally tested to maintain a realistic scenario from the initial screen to the final screen that retrieves data from records. Verbal description of the suspect's face in photo search and sketch in photo search is essential in the investigation.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest

REFERENCES

[1] Hamed Kiani Galoogahi and Terence Sim, “Face Sketch Recognition By Local Radon Binary Pattern: LRBP”, 19th IEEE International Conference on Image Processing, 2012.

[2] Charlie Frowd, Anna Petkovic, Kamran Nawaz and Yasmeen Bashir, “Automating the Processes Involved in Facial Composite Production and Identification” Symposium on Bio-inspired Learning and Intelligent Systems for Security, 2009.

[3] W. Zhang, X. Wang and X. Tang, “Coupled information theoretic encoding for face photo-sketch recognition”, in Proc. of CVPR, pp. 513-520, 2011.

[4] X. Tang and X. Wang, “Face sketch synthesis and recognition”, in Proc. of ECCV, pp. 687-694, 2003.

[5] X. Tang and X. Wang, “Face sketch recognition”, IEEE Trans. Circuits and Systems for Video Technology, vol. 14, no. 1, pp. 50-57, 2004.

[6] B. Klare and A. Jain, “Sketch to photo matching: a feature based approach”, SPIE Conference on Biometric Technology for Human Identification, 2010.

[7] Q. Liu, X. Tang, H. Jin, H. Lu, and S. Ma, “A nonlinear approach for face sketch synthesis and recognition,” Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 1005–1010, June 2005.

[8] P. Yuen and C. Man, “Human face image searching system using sketches,” IEEE Trans. SMC, Part A: Systems and Humans, vol. 37, pp. 493–504, July 2007.

[9] H. Han, B. Klare, K. Bonnen, and A. Jain, “Matching composite sketches to face photos: A component-based approach,” IEEE Trans. on Information Forensics and Security, vol.8, pp. 191–204, January 2013.

[10] P. Jeffrey Brantingham, G. E. Tita, and G. Mohler, “Gang-related crime in Los Angeles remained stable following Covid-19 social distancing orders,” Criminology and Public Policy, vol. 20, no. 3, pp. 423–436, 2021.

[11] K. Yang et al., “Benchmarking commercial emotion detection systems using realistic distortions of facial image datasets,” The Visual Computer, vol. 37, no. 6, pp. 1447-1466, 2021.

[12] L. Capozzi, J. R. Pinto, J. S. Cardoso, and A. Rebelo, “End-to-End Deep Sketch-to-Photo Matching Enforcing Realistic Photo Generation,” in Proc. of the Iberoamerican Congress on Pattern Recognition, 2021, pp 451–460.

[13] S. P. Singh and P. Singh, “An Integrated AFS-Based SWOT Analysis Approach for Evaluation of Strategies Under MCDM Environment”, Journal of Operations and Strategic Planning, vol 1, no 2, pp 129–147, 2018.

[14] S. P. Singh, M. K. Chauhan, and P. Singh, “Using multicriteria futuristic fuzzy decision hierarchy in SWOT analysis: an application in tourism industry,” International Journal of Operations Research and Information Systems, vol. 6, no. 4, pp. 38–56, 2015.

[15] S. P. Singh and P. Singh, “A hybrid decision support model using axiomatic fuzzy set theory in AHP and TOPSIS for multicriteria route selection,” Complex & Intelligent Systems, vol. 4, no. 2, pp. 133–143, 2018.

[16] K. Ounachad, M. Oualla, A. Souhar, and A. Sadiq, “Face sketch recognition-an overview,” in Proc. of the 3rd International Conference on Networking, Information Systems & Security, 2020, pp 1–8.

[17] R. Hopman and A. M’charek, “Facing the unknown suspect: Forensic DNA phenotyping and the oscillation between the individual and the collective,” BioSocieties, vol 15, no 3, pp 438–462, 2020.

[18] M. A. Khan and A. S. Jalal, “Suspect Identification using Local Facial Attributed by Fusing Facial Landmarks on the Forensic Sketch,” in Proc. of the International Conference on Contemporary Computing and Applications (IC3A), 2020, pp 181–186.

[19] M. A. Khan and A. S. Jalal, “A framework for suspect face retrieval using linguistic descriptions,”, xpert Systems with Applications, vol 141, 2020.