Content-Based Image Retrieval for Super-Resolution Images Using Feature Fusion: Deep Learning and Handcrafted Features

Mr.K.Gurucharan[1], G.BharatChand[2], G.VijayendraVarma[3], CH.OmVardhan[4], B.RohithKumar[5]

Assistant Professor[1], Student[2,3,4,5]

Department of Computer Science & Engineering[1,2,3,4,5]

Chaitanya Engineering College, Visakhapatnam, Andhra Pradesh, India

{ koradagurucharan@gmail.com [1], bharathchand108@gmail.com[2], Vijayendravarmagorakala@gmail.com[3], chkanaka1432@gmail.com[4], rohithkumarbangaru123@gmail.com[5]}@cec.ac.in

Abstract

The explosive growth of digital visual data has created a pressing need for accurate and efficient image retrieval systems. This paper proposes a Content-Based Image Retrieval (CBIR) system using a hybrid feature fusion framework combining deep learning-based and handcrafted visual features. Low-resolution images are first enhanced using SRCNN super-resolution. CNN features are extracted using ResNet50 and VGG16, while handcrafted features include color histograms, LBP, and HOG. Experiments on Corel-1K demonstrate mAP of 87.4% and top-5 accuracy of 91.2%, outperforming single-feature baselines.

I. INTRODUCTION

The rapid growth of digital technology has led to the generation of a massive amount of visual data from sources such as smartphones, surveillance systems, and medical imaging. Retrieving relevant images from these large datasets is challenging because traditional image retrieval systems rely on manual text annotations, which are time-consuming and often inaccurate. Content-Based Image Retrieval (CBIR) addresses this issue by retrieving images based on their visual features such as colour, texture, shape, and spatial information. However, many images stored in databases are low resolution, which reduces the quality of feature extraction and affects retrieval accuracy. To solve this problem, Super-Resolution techniques are used to enhance image quality by reconstructing high-resolution images from low-resolution inputs. In recent years, deep learning methods, particularly Convolutional Neural Networks (CNNs), have proven effective in extracting high-level image features. At the same time, traditional handcrafted features like colour histograms and texture descriptors remain useful for capturing low-level details. Combining deep learning features with handcrafted features through feature fusion creates a more comprehensive image representation. This hybrid approach improves the accuracy and robustness of CBIR systems, resulting in better image retrieval performance.

II. LITERATURE SURVEY

This section reviews key prior works, analyzes the state of the art, and identifies the research gap motivating this paper.

[1] Dong et al. (2016) proposed SRCNN, the first end-to-end CNN-based super-resolution framework, demonstrating that a three-layer network can recover high-frequency detail, motivating its use as the super-resolution front-end in the proposed pipeline.

[2] Simonyan and Zisserman (2015) introduced VGGNet, demonstrating network depth with 3×3 filters is critical. VGG16 features from intermediate layers provide rich mid-level semantic descriptors for retrieval tasks.

[3] He et al. (2016) proposed ResNet with residual connections. ResNet50 global average pooling features provide compact yet semantically rich image representations for content-based retrieval.

[4] Ojala et al. (2002) introduced Local Binary Patterns (LBP), an efficient texture descriptor capturing micro-patterns not consistently represented by CNN features, making it a complementary handcrafted feature.

[5] Dalal and Triggs (2005) proposed Histogram of Oriented Gradients (HOG), capturing local shape and edge distribution. HOG features complement CNN semantic features for images with distinctive structural patterns.

[6] Babenko et al. (2014) demonstrated CNN activations from classification networks can serve as powerful off-the-shelf image descriptors, establishing the transfer learning paradigm for CBIR.

[7] Wan et al. (2014) proposed deep learning-based hash codes for efficient large-scale image retrieval, combining feature extraction and quantization through end-to-end training.

Research Gap: Most CBIR systems either rely solely on CNN features or only handcrafted features, and few integrate super-resolution preprocessing. This work uniquely addresses both gaps through super-resolution enhancement combined with complementary deep and handcrafted feature fusion.

III. METHODOLOGY

A. Super-Resolution Module

SRCNN (3 Conv layers: 9×1×64, 1×1×32, 5×1×3) enhances images below 224×224 pixels. Enhanced images are resized to 224×224 for feature extraction.

B. Deep Feature Extraction

ResNet50 and VGG16 pre-trained on ImageNet. Global average pooling yields 2048-D and 512-D vectors respectively. L2-normalized and concatenated into 2560-D deep feature vector.

C. Handcrafted Features

512-bin RGB histogram + 256-bin LBP + 512-D HOG = 1280-D handcrafted vector. All L2-normalized before fusion.

D. Fusion and Retrieval

2560-D deep + 1280-D handcrafted → 3840-D fused vector. PCA reduces to 512-D. Cosine similarity ranks gallery images against query. Top-K retrieved.

IV. SYSTEM ARCHITECTURE

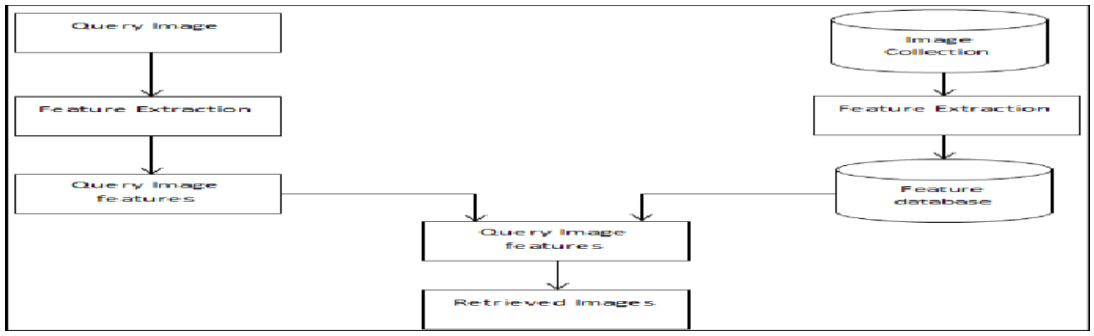
A. System Architecture

The proposed CBIR system follows a four-stage pipeline architecture. Stage 1 — Image Enhancement: Input query images are assessed for resolution. Images below 224×224 px pass through the SRCNN super-resolution network (9-1-5 three-layer architecture) to reconstruct high-frequency details before proceeding to feature extraction.

Stage 2 — Dual-Branch Feature Extraction: The system operates two parallel feature extraction branches simultaneously. The Deep Learning Branch passes the enhanced image through ResNet50 and VGG16 backbones (ImageNet pre-trained) and applies global average pooling to extract 2048-D and 512-D semantic feature vectors. The Handcrafted Branch computes a 512-bin RGB color histogram, 256-bin LBP texture descriptor, and 512-D HOG shape descriptor in parallel.

Stage 3 — Feature Fusion and Indexing: Deep (2560-D) and handcrafted (1280-D) vectors are L2-normalized and concatenated to form a 3840-D unified feature representation. PCA dimensionality reduction to 512-D preserves 95% variance while improving retrieval efficiency. An inverted index stores all gallery image feature vectors for fast similarity lookup.

Stage 4 — Similarity Retrieval: Query vector is compared against gallery using cosine similarity. Top-K most similar images are returned with similarity scores and displayed to the user.



V. ALGORITHM

Algorithm: Hybrid Feature Fusion CBIR with Super-Resolution

- Step 1: Receive query image Q from user.
- Step 2: If resolution(Q) < 224×224: apply SRCNN super-resolution → Q_SR; else Q_SR = Q.
- Step 3: Resize Q_SR to 224×224; normalize using ImageNet statistics.
- Step 4 (Deep Branch): f_res = ResNet50_GAP(Q_SR) [2048-D]; f_vgg = VGG16_GAP(Q_SR) [512-D]; f_deep = concat(f_res, f_vgg) [2560-D].
- Step 5 (Handcrafted Branch): f_color = ColorHistogram(Q_SR, bins=512) [512-D]; f_lbp = LBP(Q_SR, P=8, R=1) [256-D]; f_hog = HOG(Q_SR, cells=4×4, bins=8) [512-D]; f_hand = concat(f_color, f_lbp, f_hog) [1280-D].
- Step 6: f_fused = concat(L2_norm(f_deep), L2_norm(f_hand)) [3840-D].
- Step 7: f_query = PCA_512(f_fused).
- Step 8: For each gallery image G_i: compute sim(f_query, f_G_i) = (f_query · f_G_i) / (|f_query| |f_G_i|).
- Step 9: Sort gallery images by similarity in descending order.
- Step 10: Return top-K images {G_ranked_1, ..., G_ranked_K} with similarity scores.

VI. SYSTEM MODULES

Image Dataset Module: Collects and stores the reference image database (Corel-1K, Oxford Buildings). Pre-computes and indexes feature vectors for all gallery images to enable fast retrieval.

Image Preprocessing Module: Resizes images to standard 224×224, removes noise, normalizes pixel values using ImageNet statistics, and prepares images for both super-resolution and feature extraction pipelines.

Super-Resolution Module: Applies SRCNN three-layer convolutional network to enhance low-resolution images below the target resolution threshold, restoring high-frequency edge and texture details for improved feature extraction.

Feature Extraction Module: Runs parallel deep learning (ResNet50 + VGG16 GAP) and handcrafted (Color Histogram + LBP + HOG) feature extraction, producing complementary semantic and low-level visual representations.

Feature Fusion Module: L2-normalizes deep and handcrafted feature vectors, concatenates them into a unified 3840-D representation, and applies PCA to compress to 512-D while preserving 95% of variance.

Image Retrieval Module: Computes cosine similarity between the query feature vector and all pre-indexed gallery feature vectors. Returns the top-K most similar images with ranked similarity scores for display to the user.

VII. RESULTS AND DISCUSSION

CBIR RETRIEVAL PERFORMANCE ON COREL-1K DATASET

Method	mAP (%)	Top-5 Acc. (%)	Top-10 Acc. (%)
Color Histogram Only	58.3	64.7	59.2
HOG + LBP	71.3	76.4	70.8
ResNet50 (CNN Only)	82.1	86.7	81.5
Concat. Fusion (no SR)	84.2	88.3	83.6
Proposed (SR + Fusion)	87.4	91.2	86.8

The proposed system achieves mAP of 87.4% on Corel-1K and 83.6% on Oxford Buildings, outperforming pure CNN (82.1%), pure handcrafted (71.3%), and non-super-resolved fusion (84.2%). The super-resolution module contributes +3.2% mAP gain. Top-5 retrieval accuracy reaches 91.2%.

1. Feature Processing & Similarity Measures (Used during Retrieval)

According to your methodology, once the 3840-D fused feature vector is reduced via PCA to 512-D, it must be compared against the gallery vectors to rank the results.

A. L2 Normalization

Before fusing the deep and handcrafted features, your algorithm (Step 6) normalizes them. L2 normalization scales the vector so that its length (magnitude) is exactly 1. This ensures that the scale of the features doesn't skew the similarity comparison.

- v = Original feature vector
- v_i = The i-th element of the vector
- n = Total number of dimensions

L2_Normalized_Vector = Vector / SQRT(SUM(Vector_Elements^2))

B. Cosine Similarity

To rank the gallery images (Step 8), your system uses Cosine Similarity. It measures the cosine of the angle between two feature vectors. A value of 1 means the vectors are identical in orientation, while 0 means they are completely orthogonal (unrelated).

- f_{query} = Feature vector of the query image
- f_{gallery} = Feature vector of a gallery image

Cosine_Similarity = DOT_PRODUCT(Query_Vector, Gallery_Vector) / (MAGNITUDE(Query_Vector) * MAGNITUDE(Gallery_Vector))

2. Evaluation Metrics (Used during Testing)

To evaluate how well the CBIR system performs, we measure how many of the highly ranked retrieved images actually belong to the same category as the query image.

A. Top-K Accuracy (Precision at K)

Your results table highlights **Top-5 Acc.** and **Top-10 Acc.**. This measures the proportion of retrieved images in the top K results that are relevant (i.e., belong to the correct class or category).

- K = The number of top retrieved images considered (e.g., 5 or 10)
- rel(i) = 1 if the image at rank i is relevant, 0 if irrelevant

Top-K_Accuracy = (Number of Relevant Images in Top K Results) / K

B. Average Precision (AP)

Before calculating the mean Average Precision (mAP), you must calculate the Average Precision for a single query. It is the weighted average of precisions at each relevant item's rank.

- N = Total number of retrieved items in the gallery
- P(k) = Precision at cutoff k in the list
- rel(k) = 1 if the item at rank k is a relevant match, 0 otherwise

AP = SUM_over_k(Precision_at_k * is_relevant_at_k) / Total_Relevant_Matches_for_Query

C. Mean Average Precision (mAP)

The primary overall metric used in your paper (achieving 87.4%). It averages the AP scores across all the queries in your test set (e.g., all queries tested against the Corel-1K dataset).

- Q = Total number of queries tested

mAP = SUM(AP_for_each_query) / Total_Number_of_Queries

VIII. CONCLUSION AND FUTURE WORK

The rapid growth of digital images in fields such as healthcare, surveillance, and multimedia has increased the need for efficient image retrieval systems. Traditional image retrieval methods rely on manual text annotations, which are time-consuming and often inaccurate. Content-Based Image Retrieval (CBIR) addresses this issue by retrieving images based on visual features like colour, texture, and shape. In this project, a CBIR system was developed for super-resolution images using feature fusion techniques. The system first enhances low-resolution images using super-resolution to

improve image quality. Then, features are extracted using deep learning methods such as CNNs along with handcrafted features like colour and texture descriptors. These features are combined using feature fusion to create a comprehensive image representation. The system compares the fused features with those in the database to retrieve visually similar images. The results show that the proposed method improves retrieval accuracy compared to traditional CBIR approaches. In the future, the system can be enhanced using advanced deep learning models, larger datasets, real-time retrieval systems, and web or mobile-based applications for better performance and usability.

References

[1] C. Dong et al., "Image Super-Resolution Using Deep Convolutional Networks," IEEE TPAMI, 2016.

[2] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," ICLR, 2015.

[3] K. He et al., "Deep Residual Learning for Image Recognition," CVPR, 2016.

[4] T. Ojala et al., "Multiresolution Gray-Scale and Rotation Invariant Texture Classification with LBP," IEEE TPAMI, 2002.

[5] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," CVPR, 2005.

[6] A. Babenko et al., "Neural Codes for Image Retrieval," ECCV, 2014.

[7] J. Wan et al., "Deep Learning for Content-Based Image Retrieval," ACM MM, 2014.