Image Retrieval Using Feature Descriptors

Assignment Report

EE673: Deep Learning for Computer Vision

Full Name	Enrollment No.
Anshika	2021CSB1069
Komal	2024EEZ0013

1 | Introduction

2 | Dataset

Corel Dataset with 5 classes is used.

The five classes are :

- Bus
- Dinosaurs
- Elephants
- Flowers
- Mountains and Snow

Each class contains 30 samples.

3 | Feature Descriptors Description

3.1. Local Binary Pattern (LBP)

- **Algorithm**: Computes binary patterns by comparing each pixel's neighbours to the centre pixel, encoding a binary code based on whether each neighbour is greater or lesser than the centre pixel
- **Strength**: Efficient, captures texture information well, and is commonly used for texture classification.
- **Histogram**: Uniform LBP codes (patterns with at most 2 transitions) are given importance, while non-uniform patterns are binned together.
- **Speed**: Relatively fast because it uses built-in library functions.

3.2. Uniform LBP (Optimized)

- **Algorithm**: Similar to LBP but only considers uniform patterns (binary patterns with at most 2 transitions between 0 and 1), reducing the number of patterns significantly. Downsamples the image and reduces the number of neighbours to 4 for faster processing.
- **Strength**: Faster and more efficient than standard LBP, focuses on uniform patterns which are more meaningful for texture analysis.
- **Histogram**: Includes one extra bin for non-uniform patterns, and it's normalized like the LBP histogram.
- **Speed**: Faster due to the reduced number of neighbours and the focus on uniform patterns.

3.3. Local Mean-based Extended Binary Pattern (LMeBP)

- **Algorithm**: Similar to LBP but instead of comparing neighbours directly to the centre pixel, it compares them to the mean value of the neighbours, encoding a binary code based on this comparison. The **binary codes' 4** most significant bit planes (MSB) are used for the feature extraction.
- **Strength**: More robust to noise as the local mean comparison is less sensitive to small pixel intensity variations. Also captures more information by using the top 4 MSB.

- **Histogram**: Calculated from the binary codes' 4 MSB bit planes, focusing on significant texture patterns.
- **Speed**: Faster than U2 due to less computation

3.4. Local Median-based Extended Binary Pattern (LMeBP-Median)

- **Algorithm**: Similar to LMeBP but replaces the local mean comparison with a local median comparison. The local median of the neighbours is used to encode the binary code.
- **Strength**: Robust to outliers and noise than LMeBP, as the median is less influenced by extreme values.
- **Histogram**: Like LMeBP, it focuses on the 4 MSB of the binary codes for histogram calculation.
- **Speed**: Slightly slower due to the additional computational cost of finding the median compared to the mean.

3.5. Local Ternary Pattern (LTP)

- **Algorithm:** Uses third state in addition to the binary states. Instead of just comparing whether neighbouring pixels are greater than or less than the centre pixel, LTP introduces a threshold t to compare the neighbouring pixels as greater than, less than, or equal to the centre pixel. This results in ternary codes instead of binary.
- **Strength**: More robust to noise compared to LBP, as small intensity variations (within the threshold t) are ignored. LTP creates a ternary code by dividing pixel comparison into three categories: 1 for pixels significantly greater than the centre, -1 for pixels significantly smaller, and 0 for pixels within a tolerance threshold.
- **Histogram**: The ternary code is usually split into two binary patterns (positive and negative) for computational simplicity, and histograms are computed for each.
- **Speed**: Slightly slower than LBP due to the additional threshold-based comparison, but more robust against noise.

3.6. LBP HoG

1. **LBP Feature Extraction**:
    - The image is split into three color channels (B, G, R). For each channel and each radius LBP is calculated using the nearest neighbors, capturing local texture.
    - The LBP histogram is computed and normalized, providing rotation-invariant texture information at different scales.
2. **HOG Feature Extraction**:
    - The image is converted to grayscale. HOG extracts gradient orientation histograms from cells of size 8x8 pixels, 9 bins. This captures edge and shape information across the image.
3. **Feature Combination**:
    - The LBP histograms and HOG features are concatenated into a single feature vector for further use in classification tasks.

Speed: The computational complexity of the function depends on image size, number of LBP radii, and HOG parameters. LBP operates on every pixel for each radius, while HOG computes gradients in cells. For small-to-medium images, the method is efficient, but it slows down with larger images due to HOG's complexity.

Histograms:

- LBP histograms capture texture, with n_points + 2 bins representing uniform and non-uniform patterns.
- HOG histograms describe edge directions, with a set number of bins per orientation.

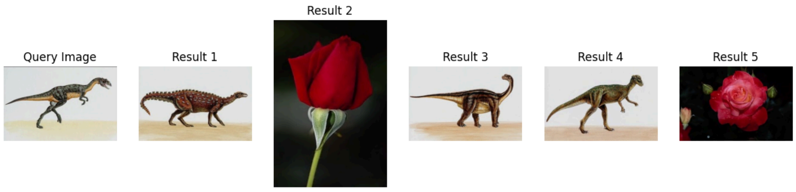
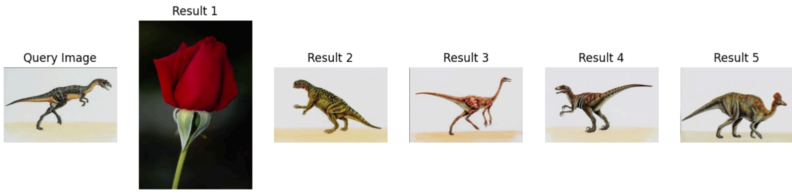
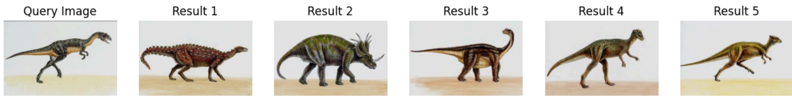
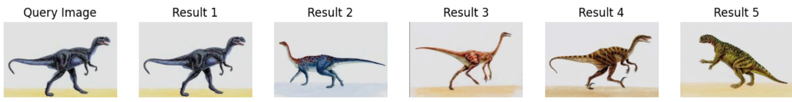
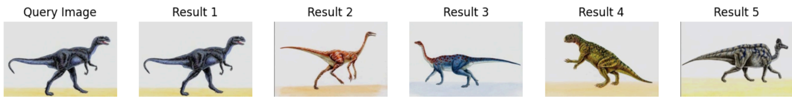
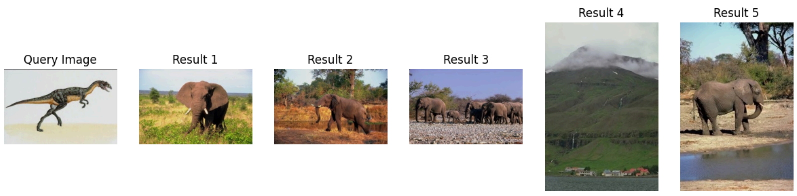
Strengths: This method captures both fine-grained texture (LBP) and shape/edge information (HOG), providing a complementary feature set. It's robust for tasks like face recognition and object detection due to its rotation invariance and multi-scale texture analysis.

4 | Feature Descriptors Results on Different Classes

4.1 | Flowers

LBP	
LTP	
Opt. U2	
Mean-LBP	
Median-LBP	
LBP HoG	

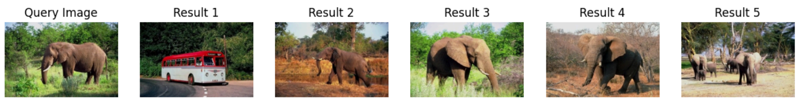
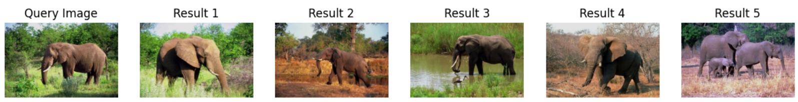
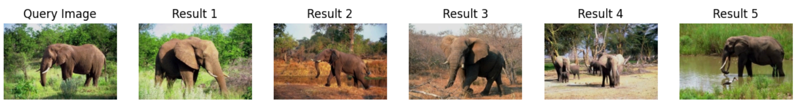
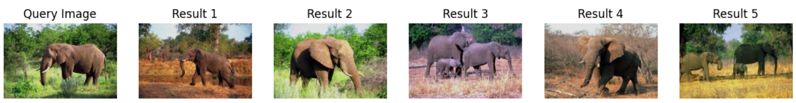
4.2 | Dinosaurs

LBP	
LTP	
Opt. U2	
Mean-LBP	
Median-LBP	
LBP HoG	

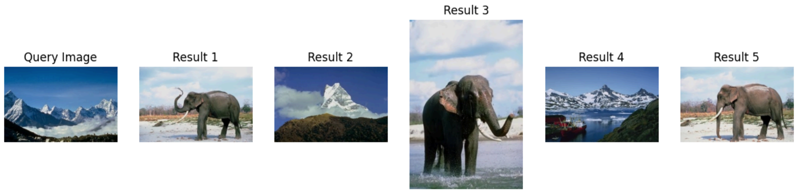
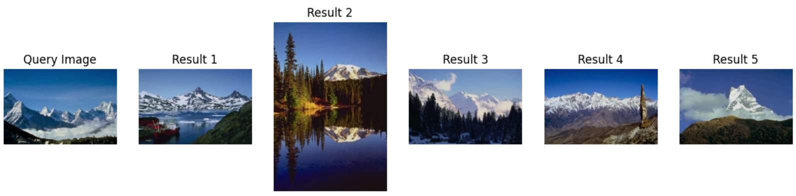
4.3 | Bus

LBP	
LTP	
Opt. U2	
Mean-LBP	
Median-LBP	
LBP HoG	

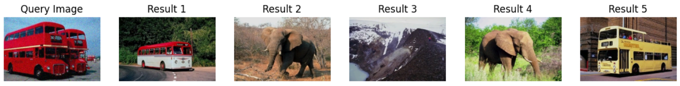
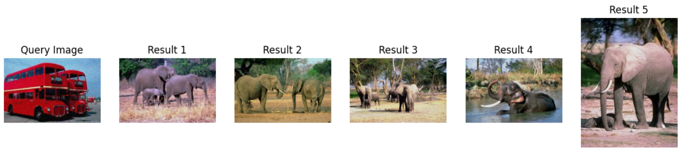
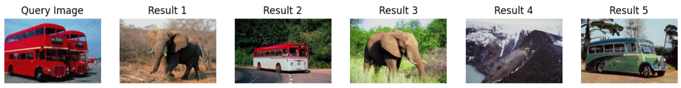
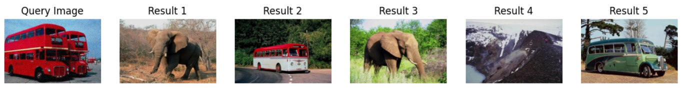
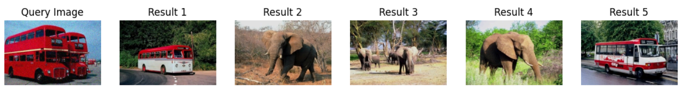
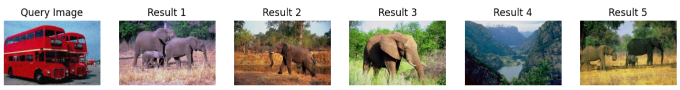
4.4 | Elephants

<table>
<tr><td>LBP</td><td>
</td></tr>
<tr><td>LTP</td><td>
</td></tr>
<tr><td>Opt. U2</td><td>
</td></tr>
<tr><td>Mean-LBP</td><td>
</td></tr>
<tr><td>Median-LBP</td><td>
</td></tr>
<tr><td>LBP HoG</td><td>
</td></tr>
</table>

4.5 | Mountains and Snow

LBP	
LTP	
Opt. U2	
Mean-LBP	
Median-LBP	
LBP HoG	

4.6 | Noisy Image Analysis

LBP		3/5
LTP		0/5
Opt. U2		3/5
Mean-LBP		3/5
Median-LBP		3/5
LBP HoG		0/5

4 | Conclusion

Descriptor	Key Concept	Strength	Robustness	Speed
Local Binary Pattern (LBP)	Compares neighbours to the centre pixel, binary encoding.	Simple, efficient, and widely used for texture.	Basic noise sensitivity.	Fast (uses built-ins).
Uniform LBP (Optimized)	Focuses on uniform patterns, downsampling for speed.	Fast, efficient, and captures important patterns.	Good robustness, faster.	Faster than LBP.
Local Ternary Pattern (LTP)	Compares neighbours using a ternary state (positive, neutral, negative)	Robust to noise with thresholding.	More robust to noise.	Slightly slower than LBP.
LMeBP	Compares neighbours to the local mean.	More robust to small variations in intensity.	More noise tolerant.	Slower than LBP due to mean calculation.
LMeBP-Median	Compares neighbours to the local median.	Robust to outliers, insensitive to extreme values.	Very robust to noise.	Slightly slower than LMeBP due to median calculation.

In this assignment, we tested the state-of-the-art feature descriptors and tested the features on Image Retrieval with Corel datasets.

We also test a few variants of these state-of-the-art feature descriptors