1. Introduction
   1. Problem and Context

There is a need for searching for semantically similar images to a given query image through a large database of images. For instance given an image of horses on a field, one maybe interested in finding all or a certain number of horse’s images in the database. This problem can be solved using a supervised learning algorithm such as deep learning. An alternative solution is to index the images based on their content. In this approach, each image of the database is represented by a set of features derived from a mathematical process such as wavelet transformation.

In this project, such as content based image retrieval (CBIR) system is designed and implemented using the so called wavelet based image indexing (WBIIS) and search proposed by Wang et al [1].

* 1. Solution used
  2. Main Findings

1. Methodology – Design – and – Implementation
   1. Software used

The implementation was done in python 2.7 along with various external packages. These packages will now be discussed.

OpenCV for image operations

OpenCV is an open source computer vision library [2]. In this work, the python version 3.4.1 was used to perform basic image operations. It was used for loading, resizing, interpolating, displaying and saving images. OpenCV 3.4.1 requires the scientific computing package numpy [3] to work.

PyWavelet for wavelet transform

To compute the wavelet transforms, the python package called PyWavelet [4] was used.

* 1. Assumptions

No prioritisation of colour variation over intensity or vis-versa

In this project, the colour variation and intensity were deemed equally important for the different queries. As such, when computing the distance between the query image and the candidate image, the corresponding weights ***wci*** were set to 1. Please refer to Wang et al for more information [1].

No prioritisation of horizontal, vertical or diagonal components

The horizontal, vertical and diagonal components of the wavelet multilevel wavelet transform were also considered equally important. Their corresponding coefficients ***W1,1, W1,2, W2,1, W2,2*** were equated to 1. For more information on the matter, please refer to [1].

Optimal Wavelet type and Multiresolution Level

Following the solution presented in Wang et al [1], Daubechies 8 wavelets and 4-layer wavelet transform were assumed optimal.

Optimal Image size

Based on Wang et al [1], we assumed that 128x128 were the optimal image rescaling dimensions.

* 1. Unimplemented Feature: Partial Query

In this work, the partial query feature was not implemented.

* 1. Design decisions and Algorithms

1. Analysis and Discussions
   1. Test Corpus Description
   2. Hardware Description

An entry level laptop computer: Asus X554L, was used for both the implementation and the test. The basic specifications of this hardware are listed below.

* Processor: Intel® Core™ i3-5010U CPU @ 2.10GHz
* RAM: 4,00 GB
* Permanent Memory: Hard disk drive
  1. Test Procedure
  2. Results

1. Conclusions
2. Recommendations
3. References

[1] O. F. James Ze Wang Gio Wiederhold and S. X. Wei, “Content-based Image Indexing and Searching Using Daubachies’ Wavelets,” *Int. J. Digit. Libr.*, vol. 1, pp. 311–328, 1997.

[2] Itseez, “Open Source Computer Vision Library.” 2015.

[3] T. Oliphant, “NumPy,” 2006. [Online]. Available: https://github.com/numpy/numpy. [Accessed: 09-Apr-2018].

[4] and C. Lee G, Wasilewski F, Gommers R, Wohlfahrt K, O’Leary A, Nahrstaedt H, “PyWavelets - Wavelet Transforms in Python.” [Online]. Available: https://github.com/PyWavelets/pywt. [Accessed: 09-Apr-2018].