Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

School of Computer Science and Statistics

Ence: An Ontology for Unstructured Still Image Data

Joshua Cassidy

September 13th, 2021

A dissertation submitted in partial fulfilment of the requirements
for the degree of
Master of Science in Computer Science

Declaration

I hereby declare that this dissertation is entirely my own work and that it has not been submitted as an exercise for a degree at this or any other university.

I have read and I understand the plagiarism provisions in the General Regulations of the University Calendar for the current year, found at http://www.tcd.ie/calendar.

I have also completed the Online Tutorial on avoiding plagiarism ‘Ready Steady Write’, located at http://tcd-ie.libguides.com/plagiarism/ready-steady-write.

Signed:_________________________ Date:_________________________

Abstract

Representing unlabelled visual data in the form of an ontology data poses a challenging problem, with existing methods first requiring the source image data to be labelled by either human annotators or by a model which relies on data labelled in accordance with a human developed taxonomy. The Ence ontology and Ogma framework are presented as a solution to this problem, the development of which are detailed in this dissertation. The Ogma framework uses a combination of existing, state-of-the-art methods and original methods to extract colours, shapes and segments from and to cluster (for the purpose of label assignment) a set of unstructured, still images in an unsupervised manner. The extracted data is then used to generate a knowledge graph from the Ence ontology. The Ence ontology facilitates images to be searched by colour, shape and similarity to other images. This dissertation outlines the design, implementation, supporting software tools, evaluation, and potential for future work of the Ogma framework and the Ence ontology. The work presented in this dissertation draws on research from the following fields: metaheuristics, unsupervised learning, genetic algorithms, ontological engineering, computer vision, machine learning, deep learning, and artificial intelligence.

Acknowledgements

I would like to thank my supervisor Professor Declan O’Sullivan for agreeing to supervise me in the development of this dissertation, for the outstanding support I have received from him and for the insightful contributions he has given me throughout the writing, research and development of this dissertation. I would like to thank Gaye Stephens for the support I received when writing this dissertation. I would like to thank Michael Walsh, Gerry O’Brian, and Joao Cabral for their setting up and giving me access to a Booles instance. I would like to thank Mohammed Sadeq Abdulhameed Al-Rawi for his insights into computer vision. I would like to thank Mr. Aidan Anthony O’Hora for his unwavering support and continued interest throughout the development of this dissertation. I would like to thank Conor Fitzsimmons for his continual motivation throughout the development of this dissertation. I would especially like to thank Thomas and Georgina Cassidy, this dissertation would not have been possible without their support.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Motivation

The current state of the art of image classification involves using labels derived from pre-existing taxonomies which are then assigned to images by human annotators or by machines. [1][2][3][4] This reliance on human annotation starts to become a major problem when working with more complex datasets, such as the CIFAR-10 dataset, an issue which is discussed in detail in chapter 3. [3][4] The heavy reliance on human annotation has given rise to an equally heavy reliance on transfer learning, which is limited to solving problems that have already been solved or that are similar to or a subset of a previously solved learning problem. Therefore, this approach is not useful with respect to the development of the system being discussed in this dissertation, a major aspect of which involves clustering images based on their contents alone, without reference to existing taxonomies which have been developed by humans. This reliance on transfer learning has resulted in the state of the art appearing to perform far better than it should be, as the models created through this method have always already been exposed to similar data, making it difficult to reproduce such models in a production environment. The ubiquity of transfer learning has also created computational power related barriers to entry, as transfer learning neural network architectures are often very complex and take a long time to train. [1][6]

A major motivation behind this dissertation is to avoid the issues described above by providing a framework (Ogma) that will allow for the semi-autonomous generation of an knowledge graph (using the Ence Ontology) from a set of unstructured, non-moving images (but not fully autonomously, as aliases for label names can be inserted into the ontology). This Ence ontology will represent the information from the unlabelled still set of images in the form of a knowledge graph in which they are explained and modelled effectively. The aforementioned Ogma framework will contain multiple sub-frameworks that will work together to extract knowledge from the still images being used to populate the Ence ontology. [1] These sub-frameworks work in tandem to form a pipeline. These sub-framework components include: labelling, color extraction, image segmentation, object detection, shape detection and image comparison.

Another aspect of the motivation for this dissertation is addressing the gap in the research into ontological development for image data (see section 3.2). Most development in ontologies goes towards creating ontologies based on tabular data and textual data, with very little research having gone into creating ontologies of visual data [7]. This is likely partly due to the fact that visual data can be difficult to represent in an ontology as doing so presents an inverse problem, in the sense that we are trying to explain the existence and state of being of an object while only being given the experience of the object [8].

In summary, the motivation behind this dissertation is to produce an ontology (Ecne) that, given a dataset of unstructured, still images, will provide a description of these images that does not rely on subjective human image annotation for labelling or determination of colour data (with image labels being generated semi-autonomously. This ontology will be generated by the Ogma framework, which itself will be composed of sub-frameworks, each of which will be discussed in more detail in section 1.1 of this chapter and in the chapters ahead. This framework is also to extract segments and shapes from the images. It is envisaged that the Ence ontology will be reusable by other researchers and practitioners to process images in a standard ontological manner. [8]

1.2 Research Question, Objectives, and Scope

1.2.1 Research Question

This dissertation addresses a number of research questions, primarily:

To what extent does the Ence ontology effectively model data generated by the Ogma framework from an unstructured set of still images?

A secondary research question that this dissertation answers is:

Is the Ogma framework able to effectively label images using their contents?

Tertiary research questions that this dissertation addresses are:

Can the Ogma framework effectively extract dominant colours from images?

Can the Ogma framework effectively segment images in an unsupervised manner?

Can the Ogma framework effectively extract shapes in images in an unsupervised manner?

1.2.2 Research Objectives

The objectives of this research are as follows:

- Develop a framework to semi-autonomously generate a knowledge graph using the Ence ontology for non-moving image data where the only manual process is the optional labelling of image aliases.
- Review the state of the art and identify the research gaps across fields such as image clustering, image classification, image segmentation and the development of ontologies for images.
- Develop a sub-framework of the Ogma framework to automatically assign images to their categories based on their contents
- Develop a sub-framework of the Ogma framework to automatically extract a dynamic number of colours from images
- Develop a sub-framework of the Ogma framework to automatically segment images by the contents of the image, not by an arbitrary number of segments
- Draw on existing methods within the state of the art to extract shapes from images in an unsupervised manner and incorporate this into the shape extraction sub-framework of the Ogma framework

1.2.3 Scope

The scope of this dissertation was determined by the author with consideration to time constraints and other practical limitations.

The scope of the Ence ontology is:

- To model the labels that the set of still images belongs to
- To model the dominant colours that make up a still image
- To model the shapes contained within a still image
- To model the segments in the still image
- To satisfy the competency questions defined in chapter 3 through its search functionality

The scope of the Ogma framework are:

- Effectively assign still images a label based on their contents
- Effectively extract the dominant colours from still images
- Effectively segment still images based on their contents
- Effectively extract shapes from still images

1.3 Technical Approach Methodology

- Follow the Ontology Development 101 methodology throughout the development of the Ence ontology.
- Investigate relevant literature to gain an understanding of the state of the art for the algorithms being used in the dissertation.
- Evaluate the Ence Ontology and the various Ogma sub-frameworks of the dissertation to determine their performance.

1.4 Evaluation Strategy

The Ence ontology will be evaluated using competency questions which will be answered by way of SPARQL queries. For example, searching for “red cat” will return images of cats where red is a dominant colour in the image. The Ogma frameworks sub-frameworks are evaluated in isolation to determine if each effectively extracts the target data from the image set. The image clustering sub-framework is evaluated using the silhouette evaluation metric and interrogating the clusters and comparing the clusters against the dataset’s ground truths to determine how well the images are clustered. The colour extraction sub-framework is evaluated using the silhouette, and the SSI evaluation metrics by comparing a version of the image where only the extracted dominant colours are used against the original image The shape extraction sub-framework is evaluated by taking a sample of the images where circles are present and evaluating this set by manual inspection. The image segmentation sub-framework is evaluated through a manual inspection of a sample of the segmented images. The image sets being used to evaluate the Ence ontology and the Ogma framework are the CIFAR-10 and Fashion-MNIST datasets.

1.5 Evaluation

The Ence ontology will be evaluated using competency questions which will be answered by way of SPARQL queries. For example, searching for “red cat” will return images of cats where red is a dominant colour in the image. The Ogma frameworks sub-frameworks are evaluated in isolation to determine if each effectively extracts the target data from the image set. The image clustering sub-framework is evaluated using the silhouette evaluation metric and interrogating the clusters and comparing the clusters against the dataset’s ground truths to determine how well the images are clustered. The colour extraction sub-framework is evaluated using the silhouette, and the SSI evaluation metrics by comparing a version of the image where only the extracted dominant colours are used against the original image The shape extraction sub-framework is evaluated by taking a sample of the images where circles are present and evaluating this set by manual inspection. The image segmentation sub-framework is evaluated through a manual inspection of a sample of the segmented images. The image sets being used to evaluate the Ence ontology and the Ogma framework are the CIFAR-10 and Fashion-MNIST datasets.

1.6 Expected Contribution

This dissertation aims to contribute to the fields of machine learning and computer vision through the creation of the Ogma framework, which extracts dominant colours from and clusters still images in an unsupervised fashion. This dissertation also aims to validate that the shape extraction and image segmentation methods (discussed in chapter 7 and chapter 6 respectively) selected for use in the Ogma framework are effective when applied to the CIFAR-10 and Fashion-MNIST datasets. The Ence Ontology is a contribution in and of itself as this provides a method for modelling unstructured still image data. Ultimately, the Ogma framework demonstrated that performing hyperparameter optimisation for clustering image data using the silhouette evaluation metric is an ineffective approach, and that a new evaluation metric will need to be used for this to work.

1.7 Definitions

Accuracy: Accuracy is an evaluation metric which calculates the number of true positives plus the number of true negatives divided by the sum of true positives, true negatives, false negatives, and false positives thus giving the overall proportion of instances that are correctly predicted by a learning algorithm. [9]

Artificial Neural Network: A Artificial Neural Network is a machine learning algorithm which is based on the biological brain. [1]

Computer Vision: Computer Vision is the research field which seeks to derive meaning from visual data. [8]

Convolutional Neural Network: Convolutional neural networks (CNN) are a type of neural network that are highly effective at learning tasks which involve visual data. [17]

Genetic Algorithms: Genetic Algorithms are a metaheuristics algorithm which aim to find the maximum or minimum of a function by evolving, genetic algorithms are not guaranteed to give the best solution to a problem but often give a good solution to an optimisation problem. [30]

Ground Truth: The ground truth in of a dataset is the expected result of a prediction i.e. the training labels in a dataset. [31]

Hyper parameterisation optimisation: A technique of automatically selecting an algorithms hyperparameters. [1]

Loss Function: A loss function is a component of a Neural Network which is responsible for evaluating the quality of the output of a Neural Network. [10]

Self-Organizing Maps (SOM): SOM is a type of artificial neural network which uses competitive learning as the training algorithm instead of backpropagation. [11]

Silhouette: The silhouette evaluation metric is a clustering evaluation metric which does not require the ground truths for the labels for data and analyses the quality of clusters with higher values corresponding to dense and well separated clusters. [12]

Structural Similarity Index: The Structural Similarity Index (SSI) calculates the amount of structural similarity between images with higher values indicating a stronger similarity between images. [13]

Scale-Invariant Feature Transform (SIFT): SIFT is a feature detection algorithm which is used to detect and match key feature points within images. [14]

Transfer Learning: Transfer learning is a technique in machine learning where a learner applies what it has learned in one setting to another similar setting. [1]

Unsupervised Learning: Unsupervised learning is a branch of machine learning where the algorithm being trained is not given a ground truth for the training data provided to the algorithm. [15]

1.8 Structure

Due to the broad scope of fields that are encountered in this dissertation, chapters 3-7 will have background sections and state of the art sections for the respective areas of computer science they discuss.

1.7.1 Chapter 1

Chapter 1 outlines the motivation, research questions, objectives, scope, technical approach, evaluation strategy and expected contribution of this dissertation.

1.7.2 Chapter 2

Chapter 2 outlines the system design of this dissertation, which includes an activity diagram of the Ogma framework, its processing of data and the libraries it uses. Also included is a class diagram of the Ence Ontology class diagram to show the structure of the Ence ontology's representation of images.

1.7.3 Chapter 3

Chapter 3 outlines the implementation, search functionality and the evaluation of the Ence ontology.

1.7.4 Chapter 4

Chapter 4 describes the clustering sub-framework of the Ogma framework. This sub-framework expands the Deep Embedded Clustering (DEC) state of the art clustering model through the application of hyperparameter optimisation, optimising for the model's silhouette evaluation metric using genetic algorithms.

1.7.5 Chapter 5

Chapter 5 outlines the colour extraction sub-framework of the Ogma framework. This sub-framework uses a self-organising map (SOM) tuned by hyperparameter optimisation, where the model silhouette evaluation metric for clustering image colour data was maximised.

1.7.6 Chapter 6

Chapter 6 outlines the image segmentation sub-framework of the Ogma framework. This sub-framework clusters features within images using convolutional neural networks, which is the current state of the art approach to image segmentation.

1.7.7 Chapter 7

Chapter 7 outlines the shape extraction sub-framework of the Ogma framework. This sub-framework utilises the circle Hough transform algorithm to extract circles from images in an unsupervised manner.

1.7.8 Chapter 8
Chapter 8 outlines the conclusions derived from this dissertation.

1.7.9 Chapter 9
Chapter 9 discusses possible future work and research that could build upon this dissertation.

2 System Design

2.1 Overview

This chapter details the design of the system, presents the flow of the system's components and introduces each of the sub-frameworks that are contained within the system.

The proposed Ence ontology will attempt to model the meaning conveyed in image data by extracting the following attributes of the images:

- The category of the image in the context of the dataset
- The shapes contained within the image
- The segments of the image (this involves breaking the image into different parts to try to explain the part of the image)
- The dominant colours within an image (i.e. a black cat lying on green grass would return the green and black colours)

Figure 2.1.1: Example image of a cat from the CIFAR-10 dataset

Figure 2.1.1 shows an image of a cat from the CIFAR-10 dataset that will be used to show the examples of shape extraction, colour extraction, and image segmentation in the following subsections.

Image Clustering Sub-Framework

The image clustering sub-framework should form clusters of images when there are enough images to group by similarity. While these grouped images will contain some differences in their data, this data will be treated as noise by the convolutional neural network, provided it does not influence it to place the image into another category (i.e. in a set of images of cats, one cat picture has writing then wall in the background, the writing would be treated as white noise.) The image labels developed for the ontology are determined by the diversity of images in the given dataset, i.e. once an outlier is encountered a new label will be created and later assigned to images encountered that are similar to the outlier. An interesting aspect of the functioning of the clustering sub-framework is that the ontology uses automated hyperparameter optimisation of clustering algorithms would be to autonomously generate a taxonomy for a dataset upon which a supervised machine learning model could be trained, as opposed to providing the model with a taxonomy subjectively defined by a human.

Shape Extraction Sub-Framework

Shapes which may be identified by the system within the images (currently only circles, which is further discussed in chapter 7). The extracted circles can be used to search the generated knowledge graph by specifying that image should contain a circle. [16][17]

Figure 2.1.2: Circle extracted from figure 2.1.2

Figure 2.1.2 shows the circle extracted from figure 2.1.1. We can see that figure 2.1.2 is only a subset of figure 2.1.1, with the borders of the image being slightly cropped out so the circle is in plain view of the image.

Image Segmentation Sub-Framework

The image segments are how the image is split up into different parts, with every part of the image being assigned to a segment.

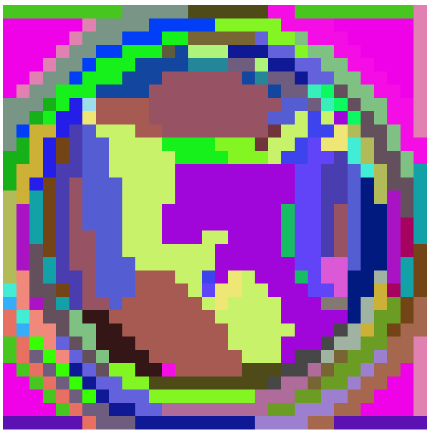


Figure 2.1.3: Figure 2.1.1 segmented

Figure 2.1.3 shows the image segments extracted from figure 2.1.1, where each segment has its own colour. We can see that the cat's shadow is being detected as its own segment. We can also see that the black borders surrounding the circle are also being picked as their own segments, while the features of the cat’s face i.e. the cat’s nose, eyes and mouth are being combined into a single segment. The cat’s ears and body are also being detected as their own segments and the shading surrounding the circle is being split into many segments (likely due to the shading of the image).

Colour Extraction Sub-Framework

The image colours are the dominant colours within an image i.e. a picture of a black cat lying on grass would ideally extract the colours black for the cat and green for the grass.

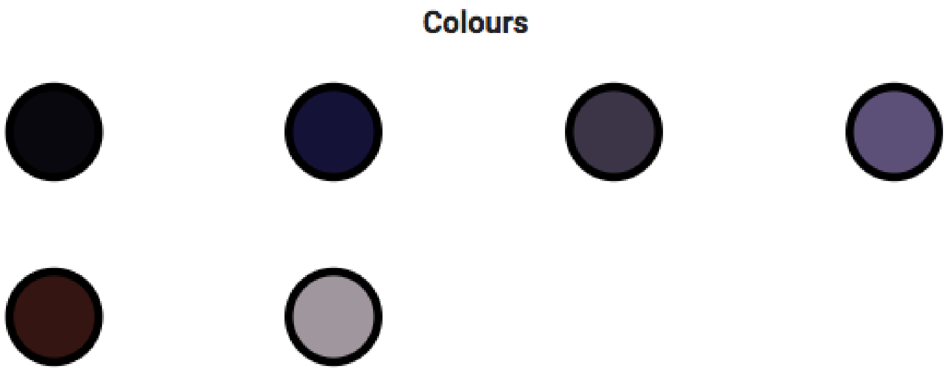


Figure 2.1.4 Colours extracted from figure 2.1.1

Figure 2.1.4 shows the colours extracted from figure 2.1.1. We can see that the colour black is extracted from the image, this being the first colour to be extracted, likely due to the black borders which make up a significant portion of the image. The dark blue colour likely derived from above the cats head, the dark gray is likely due to the cat's shadow and darker patches of fur, the next colour appears to be extracted from the cats facial features, the dark red colour appears to have been extracted from the background colour of the image of the cat and the light gray colour appears to have been extracted from the cats fur and darker portions of the circle surrounding the image.

2.2 Ence Ontological Model

This chapter details the design of the system, presents the flow of the system's components and introduces each of the sub-frameworks that are contained within the system.

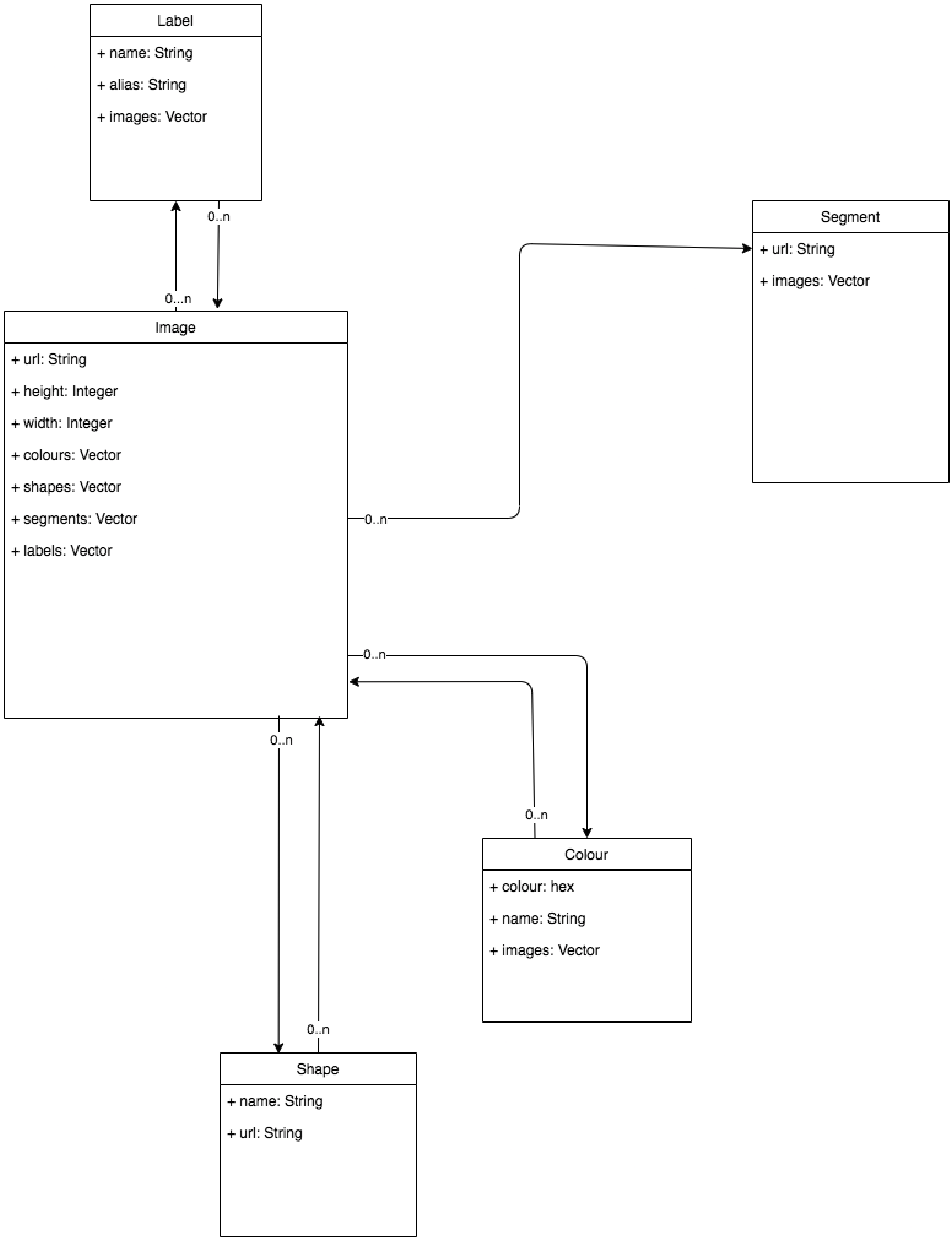


Figure 2.2.1: UML class diagram representing the Ence ontology data model

Figure 2.2.1 displays a UML class diagram representing the Ence ontology data model. From this diagram, we can see how each of the classes within the ontology relate to one another in respect to their cardinalities. The relationships between the classes are represented as properties in these classes for the purpose of presenting clearly how these relationships are organised. The structure of the ontology is detailed further in the Widico documentation provided in appendix C.

Images will be stored in the Ence ontology as data properties using strings that contain the absolute URL of the image. In Figure 2.2.1, it can be seen that the image class is at the centre of the diagram and that it is related to every other class. This represents the fact that, in the ontology, images will be associated with labels, segments, colours and shapes. To elaborate on these relationships:

- One image can have multiple labels, and labels can be associated with multiple images
- An image can have many segments attached to it, which one segment will always be attached to one image.
- An image can be related to up to 8 colours and a colour can be related to any number of images.
- An image can be related to multiple shapes and a shape can be related to multiple images

2.3 Ogma Framework

The Ogma framework consumes a set of raw still images in png, or jpg format and generates a knowledge graph using the Ence ontology using the extracted data from this set of images.

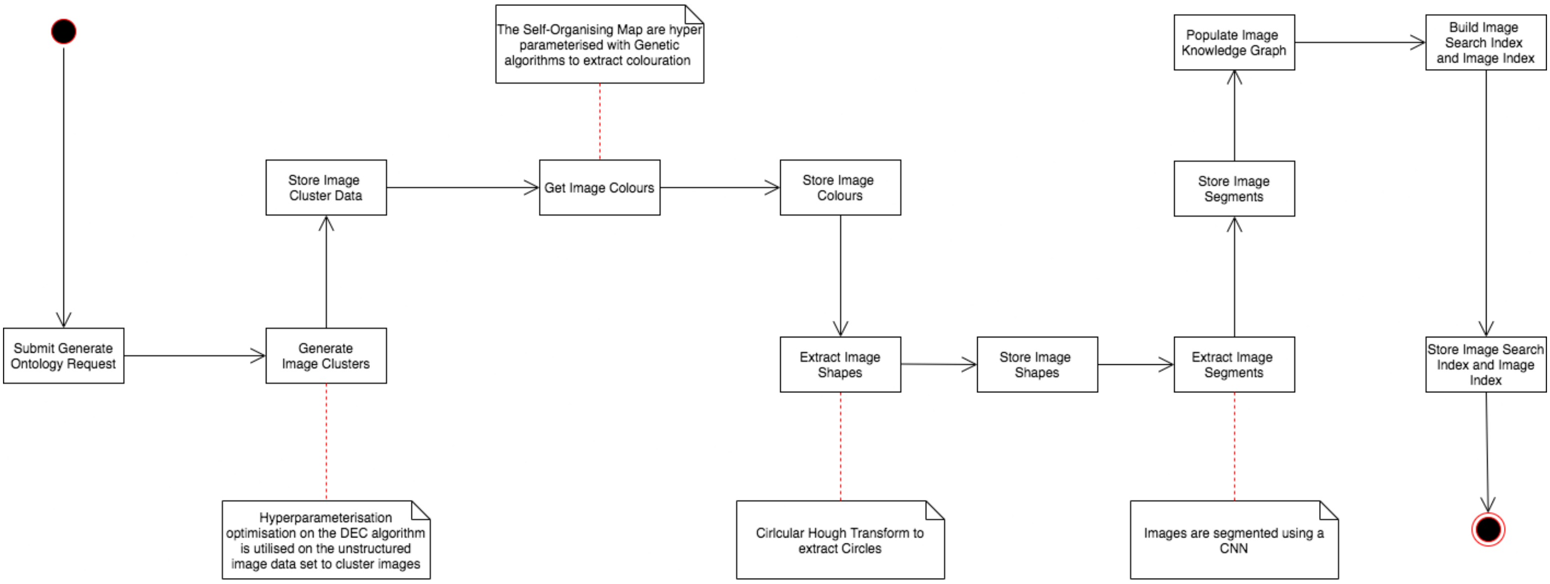


Figure 2.3.1: Activity diagram representing the Ogma framework

Figure 2.3.1 shows an activity diagram representing the Ogma framework and the end-to-end flow of its operation. From this diagram, it can be seen that the framework is essentially a pipeline connecting multiple sub-frameworks, which work together to extract data from the set of unstructured, still images to use in populating the Ence ontology.

The Ogma framework is initialised with a request to generate an ontology. The input for this request is an unstructured set of still images (more specifically, the absolute path to a directory containing these still images in the file system). This request is executed by a bash script (the general functionality of the Ogma framework's execution is driven by the Ogma.sh bash script which can be found in appendix C). The output of the Ogma framework is a knowledge graph of the Ence ontology populated by the set of images. Upon submitting the request by executing the Ogma.sh bash script, the Ogma framework will generate the image clusters using the image clustering sub-framework (detailed in chapter 3). Upon completion, the image cluster data for each of the images will be written to a CSV file, of the sub-frameworks using the python csv native library.

Image Labels CSV columns				
url	label	width	height	alias

Table 2.3.1: Image Cluster Columns CSV file

Table 2.3.1 shows the columns of the populated CSV data file upon the image clustering sub-framework completing. We can see that these columns correspond with the image class and the label class in figure 2.3.1. The column that links these two classes together is the url column, which represents the location of the image on the file system.

Following this, the images are then segmented through the image segmentation sub-framework, as described in chapter 6. The image segmentation data is also written to a CSV upon segmenting all of the images in the specified dataset.

Image Segmentation CSV columns	
url	orgin_url

Table 2.3.2*: Image Segmentation Columns CSV file*

Table 2.3.2 shows columns of the populated CSV data file upon the image segmentation sub-framework completing. The url column refers to the location on the file system of the image segments, while the origin_url column refers to the source image, which was used to link the segmentation and image class together.

Once the image segmentation sub-framework has finished, the Ogma colour extraction sub-framework (defined in chapter 5) is then executed to extract the dominant colours contained within each of the images. The extracted colours from each of the images are also stored in a CSV file.

Image Colour CSV columns						
url	colour	name	ssi	silhouette	custom_metric	image_proportion

Table 2.3.3: Image Colour Columns CSV file

Table 2.3.3 shows the columns of the populated CSV data file upon the colour extraction sub-framework completing. The url column links this data to the image class and represents the location of the image on the file system. We can see that there are additional data columns here that are not present in the image class or the colour class, these being SSI, silhouette, custom_metric, and image_proportion columns. This is because when the colours are extracted from each image, individual evaluation is performed on the source image and these metrics are stored in this CSV file for the purposes of evaluation (as discussed in chapter 5). The columns relevant with respect to populating the Ence ontology are url, colour and name (e.g. red, cornflowerblue, etc.).

When the colour extraction framework finishes, the shape extraction sub-framework executes next, extracting the circles from the images and storing the results in CSV files.

Shape Extraction CSV columns		
url	extracted_shape_url	shape

Table 2.3.4: Image Segmentation Columns CSV file

Table 2.3.4 shows the columns of the populated CSV data file upon the shape extraction sub-framework completing. The extracted_shape_url column refers to the location on the file system of the image containing the extracted shapes and url refers to the location of the source image in which the shapes were extracted (the latter being used to link the Shape and Image class together). The shape column contains the name of the extracted shape (in this case this will always be “circle”, but this column was added to support future development of the Ence ontology to support more types of shapes).

Once the image clustering, image segmentation, colour extraction and shape extraction sub-frameworks have finished executing, the Ence ontology is populated with the output of these frameworks that has been stored to CSV files. Once the Ence ontology is populated, the Non-Metric Space Library (NMSLIB) library with the Hierarchical Navigable Small World graphs (HSNW) method is used to build the image search component of the ontology using the images’ SIFT data. This search index and image index is stored to the local file system for when the Ence ontology is queried.

The effectiveness of the sub-frameworks of the Ogma framework on the test datasets were evaluated sequentially.

Library	Clustering sub-framework	Colour Extraction sub-framework	Shape Extraction sub-framework	Image Segmentation sub-framework	Ontology Search
Scikit-learn [19]	✓	✓	✕	✓	✕
Keras [20]	✓	✕	✕	✕	✕
Pytorch [21]	✕	✕	✕	✓	✕
OpenCV [22]	✓	✓	✕	✓	✓
RDFLib [23]	✕	✕	✕	✕	✕
Parallel [24]	✕	✓	✕	✕	✕
MiniSom [25]	✕	✓	✕	✕	✕
webcolors [26]	✕	✓	✕	✕	✕
NMSLIB [27]	✕	✕	✕	✕	✓
scikit-image [28]	✕	✓	✕	✕	✕
Owlready2 [29]	✕		✕	✕	✓

Table 2.3.5: Python Software Libraries used in the development of the Ence ontology search and Ogma sub-frameworks

Algorithm	Clustering sub-framework	Colour Extraction sub-framework	Shape Extraction sub-framework	Image Segmentation sub-framework	Ontology Search
SIFT [14]	✕	✕	✕	✕	✓
DEC [19]	✓	✕	✕	✕	✕
SOM [11]	✕	✓	✕	✕	✕
Genetic Algorithms [30]	✓	✓	✕	✕	✕
Circle Hough Transform [16]	✕	✕	✓	✕	✕
Image Segmentation Based on Differentiable Feature Clustering [32]	✕	✕	✕	✓	✕

Table 2.3.6: Algorithms used in the development of the Ence ontology search and Ogma sub-frameworks

Tables 2.3.5 and table 2.3.6 show the python software libraries and algorithms used in the development of the Ogma framework. Each of the Ogma sub-frameworks was developed by the author of this dissertation with the support of these libraries and algorithms, which have been selected on the basis of reflecting the state of the art in their respective fields.

2.3.1 Clustering sub-framework

The clustering sub-framework (detailed in chapter 3) used the DEC clustering algorithm to cluster the images that the algorithm was tested on. The DEC algorithm was implemented using the KERAS deep learning library by the author, and the OpenCV library was used to load the images from the file system for the training and testing of this component of the sub-framework. The DEC algorithm was implemented as specified in the Unsupervised Deep Embedding for Clustering Analysis paper, except for the number of layers and weights, which was decided using genetic algorithms with hyperparameter optimisation. To facilitate the genetic algorithms process, the code for the DEC model is generated and run by a separate python instance once the code for the model has been generated. This approach was taken for the purpose of simplicity due to the complexity of the Keras library. The genetic algorithms process was coded from scratch in python in order for it to tailor it specifically to this problem. As is later discussed in chapter 3, genetic algorithms were used to maximise the silhouette score of the model. The sci-kit learn library was used to compute the silhouette evaluation metric. These components were all combined into the clustering sub-framework, which itself exists within the Ogma framework. [12][19] [32]

2.3.2 Colour Extraction sub-framework

The colour extraction sub-framework, detailed in chapter 5, used the SOM clustering algorithm to cluster the colours within the images on which the colour extraction sub-framework was executed. The Minisom library was used for this implementation of the SOM algorithm. The OpenCV library was used to load the images that the algorithm was being executed on from the file system. The genetic algorithms process was coded from scratch to best work with the SOM model library being used. As discussed in chapter 5, the hyperparameters of the SOM algorithm that were optimised with genetic algorithms were the amount of clusters in the SOM model, its sigma value, its neighbourhood function and its topology. The genetic algorithms process' fitness evaluation function aimed to maximise the sum of the silhouette and SSI of the model. The silhouette evaluation metric in the colour extraction sub-framework was calculated using the Scikit-learn library, while the SSI evaluation metric was computed using the Scikit Image library. These metrics are used to evaluate the colour extraction framework, as discussed in chapter 5.

The webcolors library was used to determine the name of the colour that was detected from the clustering. In the event that the colour that was detected from clustering was not present in webcolors (which uses the CSS3 taxonomy of colours), the nearest colour is taken using the Euclidean distance between the colours (treating the RGB properties as a 3 dimensional space and taking the nearest colour) to give a name to the colour for the named colour search in the ontology. These components were tied together to create the clustering sub-framework of the Ogma framework. [33]

2.3.3 Shape Extraction sub-framework

As detailed in chapter 5, the shape extraction sub-framework used the Hough transform to detect circles within the images. The OpenCV library was used to load the images and to execute the hough transform algorithm on them. To automatically select the hyperparameters of the Canny shape extraction kernel in the Hough transform algorithm, the Otsu method was applied, using the OpenCV library to compute the high and low thresholds for the Hough transform algorithm. [34]
The circle Hough transform (CHT) algorithm parameters have been tuned with the aim of extracting circles that are significant to the image, with circles below a certain relative size being ignored as noise. The minimum radius of a circle contained within the image must be 10 percent of either the image's height or width (whichever is the smaller) and the minimum distance that must exist between circles centers is 3 percent of the image's height or width (again, whichever is smallest).

2.3.4 Image Segmentation sub-framework

The image segmentation sub-framework, further detailed in chapter 6, used the Image Segmentation Based on Differentiable Feature Clustering algorithm as defined in the Unsupervised Learning of Image Segmentation Based on Differentiable Feature Clustering paper. The algorithm described in this paper is a convolutional neural network based algorithm. This algorithm was implemented as defined in the aforementioned paper using the pytorch library and used as the image segmentation sub-framework for this dissertation. [32]

2.4 Testing

The code for this dissertation was tested through unit testing. These tests were developed to ensure that the low-level functions and classes that were developed for this project operated as expected, namely the functions that were core to the Ogma frameworks functionality which had output that was difficult to visualise (specifically, the genetic algorithms operators, such as the select operator, the mutation operator, etc.) The tests that were written test the code by mocking class dependencies; only the code developed is tested. A set of test cases are documented in Appendix D.

3 The Ence Ontology

3.1 Overview

This chapter outlines the design, structure and search functionality of the Ence ontology.

The Ence ontology is designed to model images based on their contents, these contents being the label of the image in the context of the dataset, the dominant colours within the image, and the segments of the image. The search component of the ontology utilises SPARQL with the NMSLIB to query the ontology and to search images by their SIFT features. There are multiple search options for the ontology, these being searching by colour, searching by shape (as described in chapter 4, the only shape that could be objectively extracted was circles by way of Hough Transforms), and searching for similar images to an uploaded image (using SIFT features). The research question that is addressed in this chapter is "*To what extent does the Ence ontology effectively model data generated by the Ogma framework from an unstructured set of still images?* " the extent in which the Ence ontology is able to model data from an unstructured set of images from data produced by the Ogma framework is quantified by the competency questions used to evaluate the ontology. The unstructured image datasets used in the evaluation of these competency questions are the Fashion-MNIST and CIFAR-10 datasets.

Competency Questions

These competency questions for evaluating the Ence Ontology are designed to highlight the capabilities of the Ence ontology model.

1. What are similar images that belong to the same category as an uploaded image?

This competency question evaluates whether the user is able to find images similar to another image which belongs to the same category within the generated knowledge graph. For example, if an image of a bag is searched other images of bags should be returned in the query results.

2. What images contain the colour "#000"?

This competency question evaluates whether the user is able to find images that contain a specific colour in this case the hexadecimal colour "#000" within the generated knowledge graph. For example ,this query should return images where #000 is a dominant colour.

3. What images contain the colour "#000" OR "#e2e2e2"?

This competency question evaluates whether the user is able to find images that contain specific colours using the logical OR operator in this case images that contain the hexadecimal colour "#000" or the hexadecimal colour "#e2e2e2" within the generated knowledge graph. For example, this query should return images where #000 is a dominant colour or where #e2e2e2 is a dominant colour.

4. What images contain the colour "#000" AND "#e2e2e2"?
This competency question evaluates whether the user is able to find images that contain specific colours using the logical AND operator (in this example, images that contain the hexadecimal colour "#000" and the hexadecimal colour "#e2e2e2") within the generated knowledge graph. For example, this query should return images where both the #000 colour and where #e2e2e2 are dominant colours.

5. What are similar images that belong to the same category as an uploaded image AND contain the colour "#000"?
This competency question evaluates whether the user is able to find images similar to another image which belongs to the same category and images that contain a specific colour (in this case the hexadecimal colour "#000") within the generated knowledge. For example if an image of a bag is being searched with the hexadecimal colour "#000" is being searched the images of bags where the colour #000 is a dominant colour should be returned in the quey results.

6. What are similar images that belong to the same category as an uploaded image AND contain the colour "#000" OR "#e2e2e2"?
This competency question evaluates whether the user is able to find images similar to another image which belong to the same category and images that contain specific colours (in this case the hexadecimal colour "#000" and the hexadecimal colour #e2e2e2) within the generated knowledge. For example, if an image of a bag is being searched with the hexadecimal colours "#000" or #e2e2e2, images of bags where the colours #000 or e2e2e2 are dominant should be returned in the query results.

7. What are similar images that belong to the same category as an uploaded image AND contain the colour "#000" AND "#e2e2e2"?
This competency question evaluates whether the user is able to find images similar to another image which belongs to the same category and images that contain a specific colour sin this case the hexadecimal colour "#000" and the hexadecimal colour #e2e2e2 within the generated knowledge. For example, if an image of a bag is being searched with the hexadecimal colours "#000" and #e2e2e2, images of bags where the colours #000 and e2e2e2 are dominant should be returned in the query results.

8. What images contain a colour that is similar to “#000”?
This competency question evaluates whether the user is able to find images that contain a similar colour to the colour that is searched. For example, a query for colours similar to #000 should result in images where colours such as #110E11 or #111 are dominant.

9. What images contain the a colour that is similar to “#F2F2F2” AND "#e2e2e2"?

This competency question evaluates whether the user is able to find images that contain similar colours to the searched colours, so in this case images that contain #FFF (which is similar to #F2F2F2) and #DFDFDF (which is similar to #E2E2E2)would be returned as search results from the generated knowledge graph.

10. What images contain the a colour that is similar to “#F2F2F2” OR "#e2e2e2"?

This competency question evaluates whether the user is able to find images that contain similar colours to the searched colours so in this case images that contain #FFF which is similar to #F2F2F2 or #DFDFDF which is similar to #E2E2E2 would be returned as search results from the generated knowledge graph.

3.2 Background

Image Ontology Background

There has been very little in the way of research into modelling the contents of images in the form of an ontology. As for what research does exist in this area, this mostly focuses on the use of captioned image datasets, while what is novel about the Ence ontology it is explicitly modelled on images that have not been captioned based on human-developed taxonomies (as explained in greater detail in the previous chapters of this dissertation). [7] The standard conventional technologies that are used to to build ontologies at the time of writing are SPARQL, OWL, RDF, RDFS and Owlready2 all of which are utilised in the creation of the Ence ontology. [29] [35]

Image Search Background

Ontology search and information retrieval in general are all very broad topics, basically revolving around returning materials from an unstructured dataset of some kind which satisfy the criteria of a specific information request. ordinarily this refers to the finding of material that is of an unstructured nature which satisfies a specific information need. [36] Content-based image retrieval is the research field that handles retrieving visually similar images from a collection of images. One method that is typically used for search would be the captioning of images then using traditional vector space models in order to perform search on the images. However, as the Ogma framework does not automatically create captions for images, in the interests of keeping the system as automated as possible and avoiding introducing human subjectivity, this is not done. [8] [36]. Feature detection algorithms are commonly used in content based retrieval with image search. These algorithms facilitate the detection and matching of key feature points within images. Examples of these algorithms are SIFT, SURF and GIST. [37] [38]

3.3 State of the Art

Image Ontology

As mentioned in the background section of this chapter there is a lack of research into the field of modelling ontologies for images, with existing research captioning the contents of images then building an ontology based on these captions.

Image Search

Deep learning methods for key feature point detection methods do not significantly outperform the classical methods of key feature point detection, which shows the sophistication of the classical methods as it is well known that deep learning has led to revolutionary changes in a variety of fields in computer science. [1] [39] [40] The GIST algorithm is highly effective at finding near duplicates in a set of images, however the SIFT algorithm is more effective at finding local image descriptors. The SIFT algorithm is more appropriate in this case, as near duplicates are not being searched in the ENCE ontology but rather images with similar features. Therefore, the SIFT algorithm is chosen over the GIST algorithm. [37]. The SIFT algorithm has been shown to be more effective overall than the other algorithms (SIFT, SURF, KAZE, BRISK and ORB) in the detection of key feature points, making the SIFT algorithm a better choice for detecting local features within images. Therefore, the SIFT algorithm was utilised for the content based image retrieval functionality. [38] [40] [41]

3.4 Implementation

The Ence ontology was modelled using Protégé to model and exported into OWL. A knowledge graph was then generated, using the Ogma framework to populate the ontology. This was conducted using the Owlready2 python software library. No external data resources were used in the creation of the knowledge graph generated by Ogma or in the Ence ontology. No data sources other than the output of the Ogma framework as used to populate the Ence ontology.

Design of the Ence Ontology

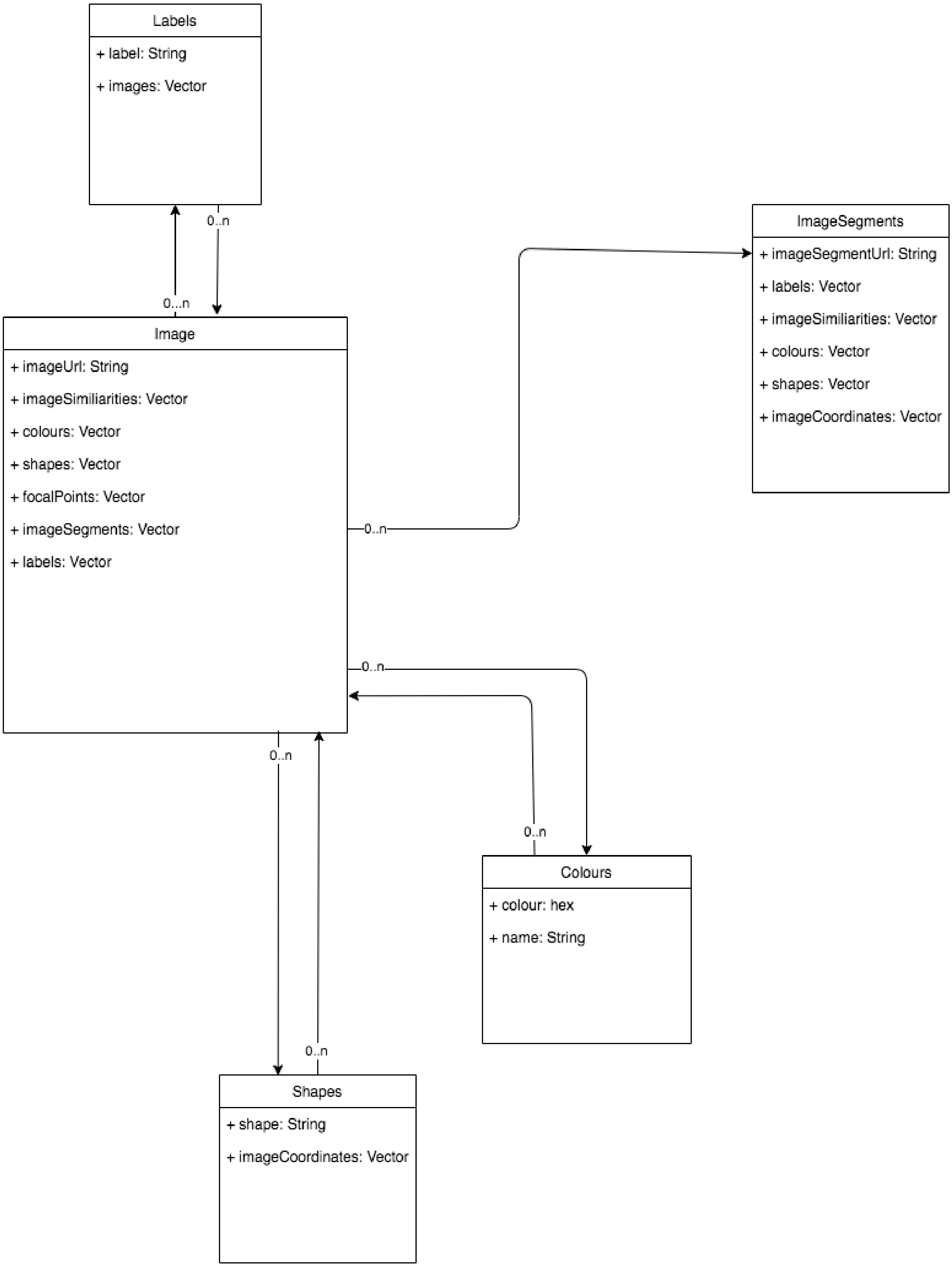


Figure 3.4.1: UML class diagram representing the Ence ontology model

The design process of the Ence ontology was conducted by researching the state of the art in image information extraction, with the aim of determining the most effective methods of doing so that would adequately describe images while also functioning within the hardware limitations of the author. SIFT features between images are rendered by the frontend in real-time as storing this data was a big data problem and is not feasible with large datasets. These SIFT features could be used to render similar SIFT features within both images, however, the datasets being used are the CIFAR-10 dataset and the fashion-MNIST datasets which contain 60,000 images and 70,000 images in total. To compute the similar SIFT features between both datasets would result in 60,000^2 and 70,000^2 combinations respectively. Therefore, the reasoning behind modelling the Ence ontology to not contain SIFT data is that ontologies have been shown to be not well suited to big data problems [43].

Mapping Process
Upon designing the Ence ontology using a UML class diagram, the Protégé software program was used to store the Ence ontology to an OWL file. All of the information contained within the UML class diagram of the Ence ontology was modelled in Protégé, including the classes, the relationships between the classes and their cardinalities, the data properties and their types, and the constraints that exist between the classes.

Transitive Relationships Between Classes
A transitive relationship between classes in an ontology where there are three classes, classes Class A, Class B, and Class C, and where there are relationships between both Class A and Class B and also Class B and Class C. In such a case, it would then be said that there is a transitive relationship between Class A and Class C that would be represented by a property linking those classes together. There are no transitive properties within the Ence ontology. [35]

Inverse Relationships Between Classes
An inverse relationship exists between two classes, class A and class B within an ontology where class A has a relationship with class B and there exists a relationship which is the inverse of this first relationship between B and A, where the relationship between A and B and between B and A are mathematical opposites of each other. [35]

Symmetric Relationships Between Classes
A symmetric relationship between classes is where there are two classes, Class A and Class B, for which there is a relationship between Class A and Class B and the same relationship between Class B and Class A. There are no symmetric properties within the Ence ontology. [35]

The Ence ontology has the following inverse relationships:

Image and Colour
The Ogma framework will only extract the 8 most dominant colours from any image. This limitation was imposed due to the fact that an image could theoretically contain as many colours as it has pixels, but if the knowledge graph stored the colour of every pixel in each image the colour search functionality would become less practically useful. For example, when searching the knowledge graph for “red” images, it would not be helpful for all images to be returned that contain even a single red pixel. Instead, it would be more useful for only images in which red is one of the 8 most dominant colours to be returned. The relationship between Image and Colour is denoted by the "hasColour" property ("hasColor" is also an alias for this property). The purpose of this property is to represent the colours that an image contains. The inverse of this property is "hasImage", this property in the context of colours and images represents colours that are present in images. For example, this property is used to searched images that contain a certain colour (#FF0000 or "red" for example).

Label and Image
An image can contain many labels in the Ence ontology, a relationship which is denoted by the "hasLabel" property. The inverse of this property is the "hasImage" property, which represents the images that are in a certain category.

Segment and Image
An image can contain many segments in the Ence ontology, a relationship which is denoted by the "hasSegment" property. The inverse of this property is the "hasImage" property, which represents the image that a specific image segment belongs to.

Shape and Image
An image can contain many shapes in the Ence ontology, a relationship which is represented by the "hasShape" property. The inverse of this property is the "hasImage" property, which represents the images that a certain shape belongs to (where one shape can be in many images).

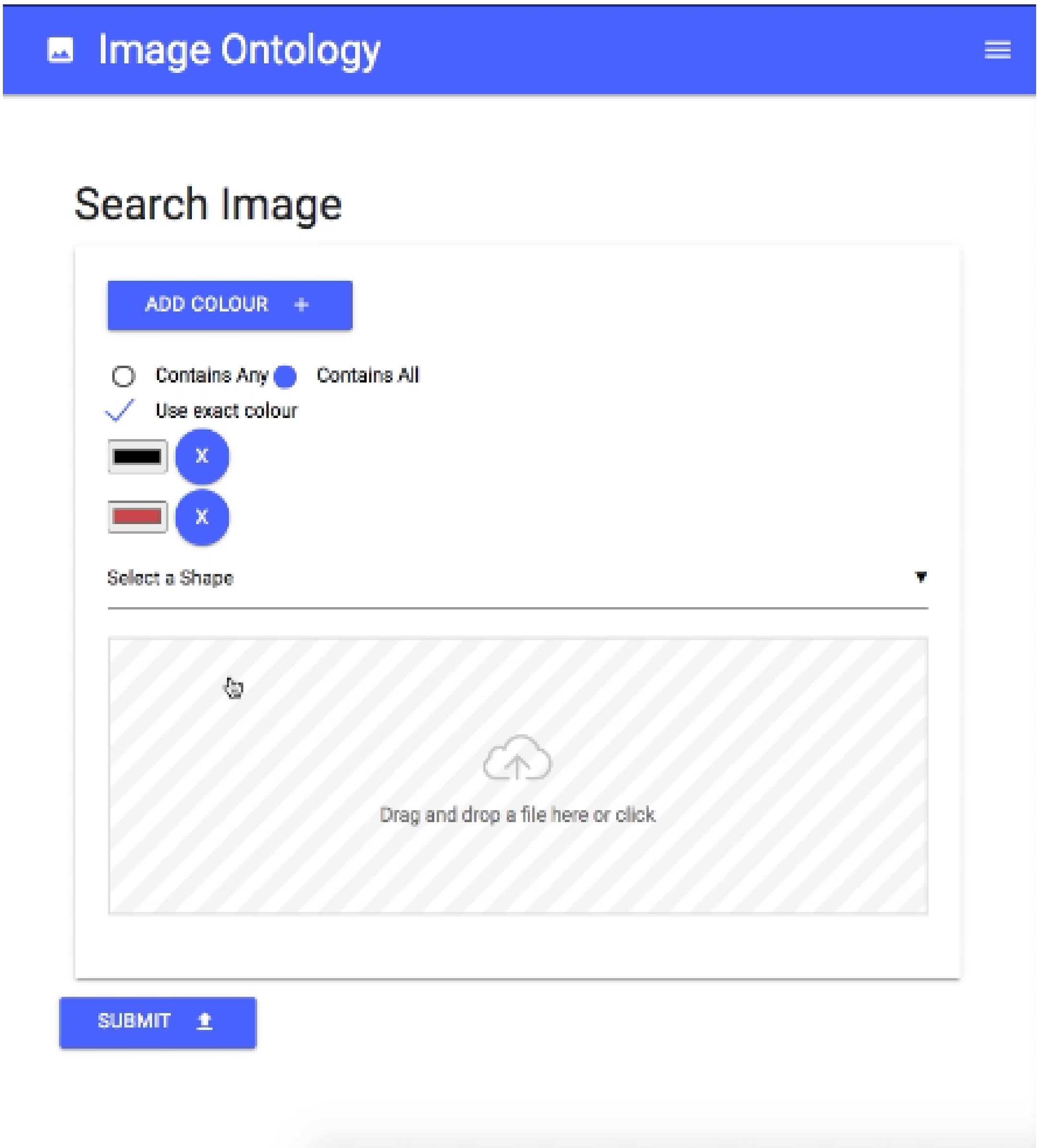
Front end


Figure 3.4.2: UI used to interact with knowledge graphs generated from the Ogma framework using the Ence ontology

Figure 3.4.2 shows the UI that is used to query the knowledge graphs generated from the Ogma framework that use the Ence ontology. This UI facilitates all of the competency questions with the exception questions 1, 6, 7 and 8, as search by category/label functionality was not included (given the findings in Chapter 3, which revealed that the clustering model did not work effectively, insofar that it assigned most images to a single label/category). For the specific implementation for the queries, please see the web UI in appendix C.

Ence Ontology Search

The competency questions 2, 3, 4, 8, 9, and 10 are implemented using SPARQL queries, with the UI shown in figure 3.4.2 facilitating these queries. Competency questions 1, 5, 6 and 7 allow for images to be searched using other images, which requires integrating SPARQL with a search index. As was indicated in the background research, the NMSLIB with the HNSW method had been shown to be highly successful when used for image search. [44] When the image is uploaded, it is not categorised using the image clustering sub-framework as described in chapter 3 due to the fact that, as previously stated, the clustering model was ineffective insofar that it categorised most images into a single category. The image search is conducted using the SIFT features within images. As shown in the background section, SIFT seems to be highly effective for extracting key points within images. The SIFT feature detection algorithm of OpenCV library was utilised with the following parameters: 100 features to be retained, 3 octave layers and a contrast threshold value of 3 and 0.09 (which has been shown to be highly effective [14]. The NMSLIB library is used to index and search the SIFT features that are extracted from the images, using the HSNW method. The reason that the NMSLIB library was utilised with HSNW is because this library is the current state of the art; it has been shown to perform well with SIFT and high dimensional data in comparison to other libraries and methods. The NMSLIB library with the HSNW method has also been shown to be highly scalable which is useful for the purposes of this analysis. [44] [45][46]

3.5 Results & Discussion

Ontology Competency Questions

The competency questions of the Ence ontology were designed to highlight the capabilities of the ontology. The knowledge graphs that were generated from the ontology used the Fashion-MNIST and CIFAR-10 datasets, and hence these datasets have been used to evaluate the competency questions.

1. What are similar images that belong to the same category as an uploaded image?

This competency question is not successfully answered by the Ence ontology as it does not facilitate category/label search. This functionality was deliberately excluded due to the ineffectiveness of Ogma framework's image clustering sub-framework (discussed under chapter

2. What images contain the colour "#000"?

This competency question is successfully answered by the Ence ontology when using the knowledge graphs generated using either the Fashion-MNIST or CIFAR-10 datasets.

3. What images contain the colour "#000" OR "#e2e2e2"?

This competency question is successfully answered by the Ence ontology when using the knowledge graphs generated using either the Fashion-MNIST or CIFAR-10 datasets.

4. What images contain the colour "#000" AND "#e2e2e2"?

This competency question is successfully answered by the Ence ontology when using the knowledge graphs generated using either the Fashion-MNIST or CIFAR-10 datasets.

5. What are similar images that belong to the same category as an uploaded image AND contain the colour "#000"?

This competency question is not successfully answered by the Ence ontology as it does not facilitate category/label search. This functionality was deliberately excluded due to the ineffectiveness of Ogma framework's image clustering sub-framework (discussed under chapter

6. What are similar images that belong to the same category as an uploaded image AND contain the colour "#000" OR "#e2e2e2"?

This competency question is not successfully answered by the Ence ontology as it does not facilitate category/label search. This functionality was deliberately excluded due to the ineffectiveness of Ogma framework's image clustering sub-framework (discussed under chapter

7. What are similar images that belong to the same category as an uploaded image AND contain the colour "#000" AND "#e2e2e2"?

This competency question is not successfully answered by the Ence ontology as it does not facilitate category/label search. This functionality was deliberately excluded due to the ineffectiveness of Ogma framework's image clustering sub-framework (discussed under chapter 4).

8. What images contain the a colour that is similar to "#F2F2F2"?

This competency question is successfully answered by the Ence ontology when using the knowledge graphs generated using either the Fashion-MNIST or CIFAR-10 datasets.

9. What images contain colours similar to "#F2F2F2" AND "#e2e2e2"?

This competency question is successfully answered by the Ence ontology when using the knowledge graphs generated using either the Fashion-MNIST or CIFAR-10 datasets.

10. What images contain colours similar to "#F2F2F2" OR "#e2e2e2"?

This competency question is successfully answered by the Ence ontology when using the knowledge graphs generated using either the Fashion-MNIST or CIFAR-10 datasets.

3.6 Conclusion & Future Work

The Ence ontology model is successful with regard to the structure of the model allowing for all the competency questions to be answered successfully, however, issues arose with the image clustering in chapter 3, which prevented competency questions 1, 5, 6 and 7 being answered successfully. The answer to the research question that this chapter addresses "*To what extent does the Ence ontology effectively model data generated by the Ogma framework from an unstructured set of still images?*" is that the Ence ontology as an ontology structure is effectively able to model colour, shape, image image segmentation data, and the image labels. However, the Ence ontology using the current iteration of the Ogma framework failed to to answer competency questions 1, 5, 6, and 7, meaning that the Ence ontology is ineffective at searching images that are the same category as an image provided due to the ineffectiveness of the clustering sub-framework discussed in chapter 3. The failure to answer these questions has to do with the data used to populate the ontology, not the ontology structure itself. The data for the shapes, segments and colours of the images are effectively produced by the Ogma framework, which means the competency questions 2 - 4 and 8 - 10 are effectively answered using the CIFAR-10 and Fashion-MNIST datasets.

The Ence ontology could be developed further to use ontology learning in a way that would be similar to how natural language ontology learning is performed, but instead using feature descriptions within the image and matching these feature descriptions to other images to show how they are related. The ontology could also be updated to contain other properties of the images, such as the spatial relations within images, the textures that are contained within images, shapes in the images other than circles (ie.g. squares, pentagons, triangles) and the ontology could also be updated to contain HSV data. This ontology could be further extended still to also accommodate different types of data other than still images, such as gifs, video data, with the sequence of the frames being analysed.

The search component of the Ence ontology could be improved by revising the clustering framework to correctly classify the images (as detailed in chapter 3), which would allow for images of the same class as the uploaded image to be requested through the search component. The search functionality could be further improved by clustering the images to allow the retrieval of similar segments that are present in the generated knowledge graph from a specific dataset, allowing for a more granular image search. Another way in which the ontology could be extended would be to treat each segment in the same manner as a source image, relating it to colours, shapes and clustering it for the purposes of labelling.

4 Image clustering Ogma Sub-framework

4.1 Overview

The sub-framework described in this chapter will apply state-of-the-art clustering techniques to still image datasets.

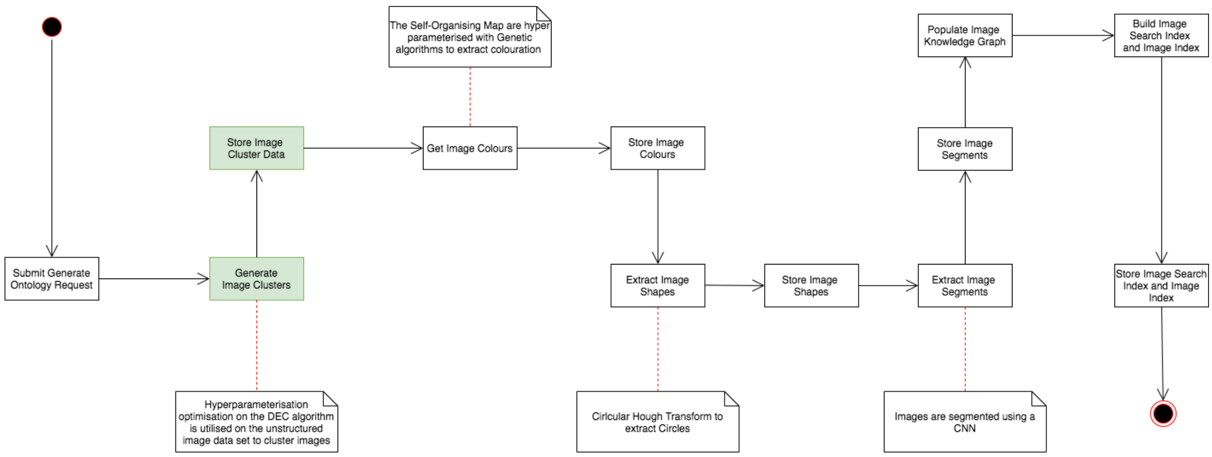


Figure 4.1.1: Activity diagram representing the Ogma framework with the Clustering processes highlighted

The research question that is addressed by the clustering sub-framework described in this chapter is "*Is the Ogma framework able to effectively label images using their contents?*". The framework uses a variable amount of clusters to cluster images, a number which is arrived at autonomously, whereas in typical approaches this number is often decided using human judgement (invariably introducing some degree of subjectivity). The number of clusters will be arrived at using hyperparameter optimisation techniques, minimising the silhouette evaluation metric.

In the traditional approaches in computer vision, images are hand labelled using pre-existing taxonomies that were developed by humans at some point in the past. However, such an approach is not ideal when using complex datasets to train models. A notable example of this is the CIFAR-10 dataset which has, at the time of writing, 12,439 citations. [3] [47]

Figure 4.1.2: Side view of Dog found in the CIFAR-10 dataset

Figure 4.1.3: Dogs face found in the CIFAR-10 dataset

In Figures 4.1.2 and 4.1.3, we can see two data instances from the CIFAR-10 dataset which are both categorised as dogs. As humans, we can see that Figure 4.1.2 shows a side view of a dog while 4.1.3 shows a dog's face, and that these are different portions of the same type of animal viewed from different angles. However, for the purposes of being used as training data for a neural network, these two images being categorised in the same way is not logical or helpful, as the visual data contained in both is actually radically different. Ultimately, a model trained on the images contained in this dataset and categorised as is (with front and side views of dogs sharing a category) may ultimately be able to identify dogs in images with some degree of accuracy, but with less accuracy than if separate categories had been created for both of these angles and the model trained to recognise side views and front views of dogs separately. However, as is commonly the case with human-annotated datasets (or datasets annotated by a machine according to a taxonomy created by humans), this issue was not accounted for in the annotation of the CIFAR-10 dataset. This is unlike the Fashion-MNIST dataset where the taxonomy is more logical with fashion items being someway at the lowest level of the data, without much wriggle room for interpretation,

This dissertation proposes that a more suitable method for using these types of complex image datasets for computer vision would be to autonomously generate the labels for the dataset based on the contents of its images using clustering with hyperparameter optimisation techniques, such that the amount of clusters selected to be used is based on the contents of the dataset. These clusters represent an autonomously generated taxonomy of images, with each cluster representing a label in the generated taxonomy. If, at a later stage, a user wanted to combine two autonomously generated labels (for example, side views and front views of dogs), it would be a simple case of using a lookup to table these labels that would be defined by the user of the dataset when the generated model is being applied.

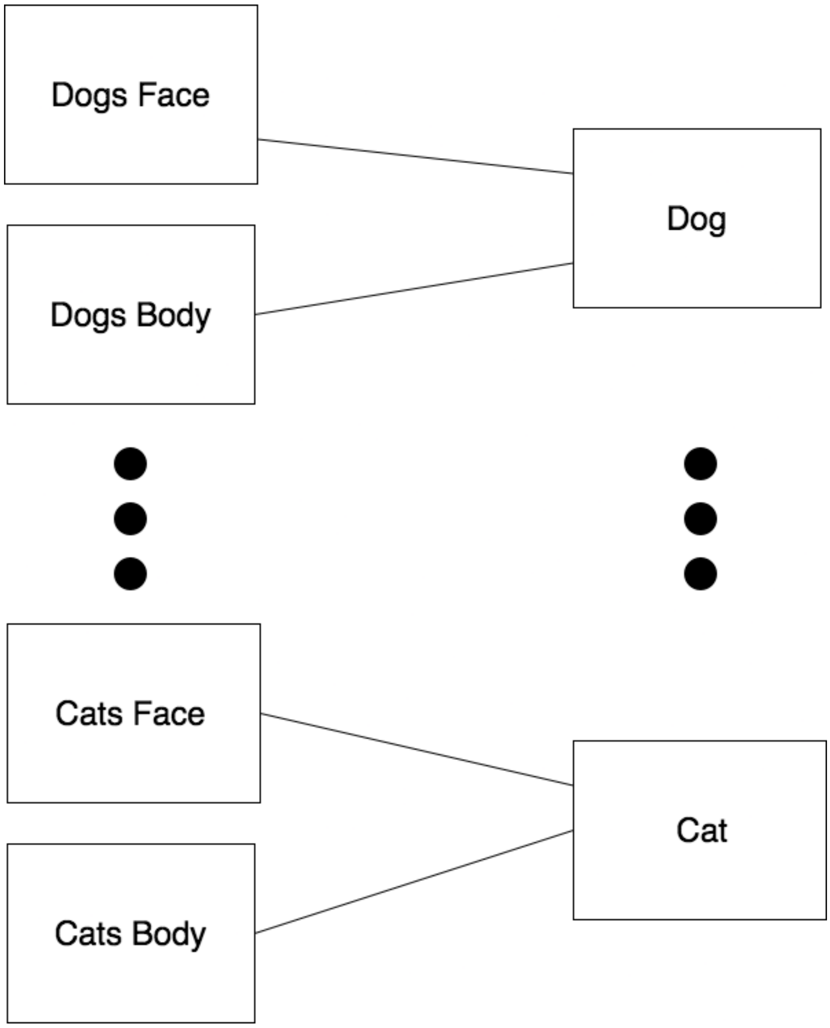


Figure 4.1.4: Generated labels of dogs and cats being mapped to dog and cat respectively

Figure 4.1.4 shows a conceptual dataset where the autonomously generated labels include: *cat faces*, *cat bodies*, *dog bodies* and *dog faces*. These labels have been, respectively, mapped to *cat* and *dog* by the user manually. The labels generated by the dataset would be represented by strings composed of numbers rather than descriptive English words (*dog face*, *cat body*, etc.), which have just been used in 4.1.4 for demonstration purposes.

The role of image clustering in the Ogma framework is to assign labels to images autonomously and based on their contents. These labels are represented by the labels class in the Ence ontology (as shown in chapter 2 figure 2.2.1). This image clustering component would ideally be used in the search mechanism of the ontology (discussed in chapter 3) so as to allow for filtering by label to exclude irrelevant images from search results (due to the poor results of the clustering framework this is not the case.

4.2 Background

Clustering

Traditionally in clustering, the number of clusters used is arrived at through a judgement call on the part of a data scientist. To avoid the subjectivity that is invariably introduced by relying on the judgement of a human, the clustering sub-framework discussed in this chapter reaches a number for clusters by applying hyperparameter optimisation using genetic algorithms to the data of the images, optimising for the silhouette value. Image clustering is still a relatively unsolved problem that is still being worked on and while there are methods that have produced very good results, many of these methods use transfer learning or pseudo labelling. Hence, these methods can be considered semi-supervised, while this chapter explores a purely unsupervised approach. Much research has been performed around clustering with respect to feature selection, grouping methods, distanced based functions, and cluster validation. [48][49][50][51][52] Some of the most popular and well-known clustering methods are k-means and gaussian mixture models, which are highly computationally efficient and are applicable to many problems, though they are known to suffer from the curse of dimensionality. [51][53][54] There have been several variants of the k-means algorithm that have been proposed in order to create a version of the k-means algorithm that performs better on high-dimensional data. One of these variants involves performing k-means clustering and dimensionality reduction on the data, maximising for inter-cluster variance. The drawback of this method was that it was limited to linear embeddings, however, the DEC algorithm expanded on this by employing deep artificial neural networks to allow for non-linear embeddings. [18][55] Spectral clustering and the variants of spectral clustering allows for more flexible distance metrics and have been used with deep autoencoders to improve performance at the expense of increasing memory consumption. [56] The DEC model, which this sub-framework is inspired by, defines a centroid based probability distribution and aims to minimise the Kullback-Leibler divergence between the auxiliary target distribution to improve the assignment of clusters and feature representation, minimizing the Kullback-Leibler between a data distribution and an embedded distribution has been used for dimensionality reduction. [18]

Dataset

The datasets that are used to evaluate the sub-framework discussed in this chapter are the CIFAR-10 dataset and the Fashion-MNIST Dataset. The CIFAR-10 dataset is a dataset created by the university of Toronto and uses a taxonomy of 10 labels (e.g. cats, dogs etc.) [uni Toronto]. This dataset contains 60,000 32x32 still colour images, with 50,000 of these belonging to the training data and 10,000 of these belonging to the test data. The test data contains 1,000 images from each category, all randomly selected. The training data has 5000 images for each category [uni Toronto]. The Fashion-MNIST was created by Zalando Research and contains 70,000 28x28 still black and white images, with 60,000 images belonging to the training data and 10,000 to the test data. The taxonomy used by this dataset is also composed of 10 categories (e.g. dresses, sandals etc.). The test data contains 1,000 images from each category, randomly selected, with the training data having 6,000 in each category [4].

Hyperparameter optimisation

Hyperparameter optimisation has been shown to be highly effective when applied to supervised learning, with respect to improving the model performance. However, there has been little research into applying hyperparameter optimisation to unsupervised methods and into image clustering methods, a research gap that this chapter aims to help to address. [57][58] The evaluation metrics that are typically used in the evaluation of image clustering are accuracy and adjusted rand index, both of these methods require us to know the ground truth for the dataset. Therefore, the approach that will be taken by this sub-framework will be to optimise the silhouette evaluation metric in the tuning of the hyperparameters. The silhouette evaluation metric uses the Manhattan distance metric instead of the Euclidean distance metric as the Manhattan distance metric has been shown to scale to high dimensions, whereas the Euclidean distance metric has been shown to perform poorly on higher dimensional data. [59]

Genetic algorithms are defined as a problem-solving technique where the program evolves to solve problems with the aim of developing a good solution, which may not be the optimal solution. [60] The theory of natural selection are what inspire how genetic algorithms operate, through the theory of natural selection, and simulating genetic operations which occur naturally which include mutation and sexual recombination. [60] Genetic algorithms arrive at their solutions by having a population of individuals reproduce over a number of generations, where each individual represents a potential solution to a problem. The members of the population are bred with one another, which lead to new population members being born which potentially possess mutations leading to the next population being made up of new members, the fitness of a population member is what determines its likelihood of reproducing. [60]

This means that the fittest members in the population are the most probable to reproduce, which results in each successive population typically improving on the previous generation, until the population converges or the system stops. [61]

The members of the population usually have fixed-length binary strings which represent their genes with new population members being created by randomly selecting single bits from their parents gene strings. Population members also experience mutations which are bits flipping within the binary genes strings. Genetic algorithms have been shown to be highly effective at solving optimisation problems [61]

4.3 State of the Art

There has been much research into the application of clustering and supervised learning for image labelling, including with using the CIFAR-10 and Fashion-MNIST datasets, both of which are balanced. The primary metric that will be used to evaluate the clustering sub-framework is the accuracy evaluation metric, as this is an evaluation metric that is commonly used in both supervised and unsupervised solutions. The state of the art in the case of the clustering sub-framework uses accuracy as the primary evaluation metric as correctly labelling images is what is considered most important. (i.e. not recall or any other evaluation metric as the amount of false negatives and positives should be minimised and the amount of true negatives and true positives should be maximised meaning that it is important that the images are correctly labelled) both the CIFAR-10 and Fashion-MNIST datasets are balanced datasets. The CIFAR-10 dataset has 10 labels which images are categorised into, 5000 images in the training data and 1000 images in the test data with the same amount of images across the 10 labels (which means that models that overfit the data will not produce artificially high scores).

The Fashion-MNIST dataset similar to the CIFAR-10 is a balanced dataset. The Fashion-MNIST datasets images are categorised into 10 labels. The Fashion-MNIST dataset has 6000 images in the training data and 1000 images in the test data with the same amount of images across the 10 labels, meaning that models that overfit the data will not produce artificially high scores. As both of these datasets are balanced this makes the accuracy evaluation metric an effective evaluation metric for the clustering sub-framework. The Ogma framework's image clustering sub-framework, while being unsupervised, aims to be more efficient than the current, state-of-the-art supervised methods relying on human developed taxonomies. Hence, these existing supervised methods will also be compared to the image clustering sub-framework. The current state of the art in supervised procedures for the CIFAR-10 dataset in terms of the accuracy evaluation metric are the LaNet, Sharpness-Aware Minimization for Efficiently Improving Generalization and EnAET, which achieve accuracies of 99.0%, 98.60%, and 99.1% respectively, meaning that each of these model are highly effect at correctly classifying the CIFAR-10 dataset. [62][63][64] The current state-of-the-art supervised learning models for the Fashion-MNIST dataset are VGG8B, NeuPDE, and Convolutional Tsetlin Machine, which achieve accuracies of 95.86%, 92.4%, and 91.4% respectively, slightly less effective results than those of the state-of-the-art models for the CIFAR-10 dataset but still highly effective. [65][66][67]

The current state-of-the-art unsupervised learning frameworks and model architectures for the CIFAR-10 dataset are the SPICE, RUC, SCAN and ConCURL model architectures, which achieve accuracy values of 91.7%, 90.3%, 88.3%, and 84.6% respectively. At first glance, these accuracy values would indicate that these model architectures would be highly effective for labelling the CIFAR-10 data in an unsupervised manner. However, upon investigating the model

architectures, frameworks and configurations, all three of these model architectures use transfer learning with differing variations of ResNet. [68]
Models that use transfer learning to cluster the data are not suitable for the Ogma framework as the clustering sub-framework aims to effectively cluster data based on its contents alone, without relying on data from a problem that has already been solved. The SPICE framework uses the WideResNet-28-2 architecture to perform transfer learning for the CIFAR-10 dataset, along with pseudo labelling, further making the SPICE framework unsuitable for use in Ogma framework, as the research question this chapter addresses specifies that the image dataset is unstructured. Similarly to the SPICE framework, the ConCURL, RUC and SCAN model architectures also use pseudo labelling. They also use the RESNET18 model architecture to perform transfer learning to cluster the CIFAR-10 dataset. The Deep Correlation Clustering Mining framework is a statistically significantly less performant image clustering framework which does not use transfer learning and has achieved an accuracy value of 62.3%. However, it does use pseudo labelling, which makes it unsuitable for the Ogma framework or the research question defined in this chapter.

The Deep Embedded Clustering (DEC) algorithm is a far less performant image clustering framework than the previously mentioned model architectures and frameworks which achieves an accuracy value of only 30.1% on the CIFAR-10 dataset. The Deep Embedded Clustering (DEC) algorithm operates by simultaneously learning the image feature representations and assigning data to clusters using a deep neural network. The DEC algorithm learns the feature mapping $f\theta : X \rightarrow Z$. The X in this equation is the feature space of the image and the Z is a lower-dimensional feature space. Upon creating this mapping the DEC algorithm minimises the Kullback-Leibler divergence value. Minimising the Kullback-Leibler divergence value minimises the internal cluster distance of the Z feature space (meaning that Z are the clusters in the DEC model). [18]

The current state-of-the-art unsupervised learning frameworks for the Fashion-MNIST dataset are the SPC (Selective Pseudo-Label Clustering), DynAE, and the model architecture created in the paper titled Semi-Unsupervised Learning: Clustering and Classifying using Ultra-Sparse Labels, which achieve accuracy values of 67.9%, 59.1%, and 90.1% respectively, values which are quite mixed. There also appears to be significantly less research in clustering the Fashion-MNIST data than there is in clustering the CIFAR-10 data, likely as the CIFAR-10 dataset was released in 2009 and the Fashion-MNIST data was released in 2017. [3][4]

The model architecture defined in the paper titled Semi-Unsupervised Learning: Clustering and Classifying using Ultra-Sparse Labels appears to be highly effective at clustering the Fashion-MNIST data, while the SPC and the DynAE are both only somewhat effective (with 70% generally being considered an effective value) at clustering the Fashion-MNIST data. All three of these models use pseudo-labelling and are thus unsuitable for the purposes of the Ogma framework. [69][70][71]

Given the unsuitability of the state-of-the-art methods outlined in this section, a different approach is taken for the clustering sub-framework of the Ogma framework. This method involves purely unsupervised methods, using this DEC with the aim of increasing its performance through hyperparameter optimisation using genetic algorithms, optimising for the silhouette evaluation metric.

4.4 Implementation

In section 4.3, it was concluded that the approach that should be taken for the image clustering sub-framework of the Ogma framework would be to use the DEC clustering model with hyperparameter optimisation using genetic algorithms. The sub-framework has been implemented using python with Keras as the deep learning framework.

The hyperparameter optimisation framework that is used to optimise the DEC model is a differential variant of genetic algorithms, which uses a mutation operator which creates children based on 3 population members and calculates the child's genes performing matrix operations to create a different child from the 3 individuals (with one individual being treated as the child's parent it will later compete against) this is done to spread the population throughout the search space efficiently as the DEC model is very computationally intensive. When a new generation is created each child is compared with the parent which created it and if the child is fitter than the parent the child will replace the parent in the population and vice versa. [72] There is also an additional mutation operator for this sub-framework’s use of genetic algorithms, which was implemented using bitflip, which entails each bit in the binary string having a low probability to be flipped (this occurs at the end of every generation). The probability that was selected in this implementation was 0.05%,, which was found to be the best when tweaking the system. [24] The differential evolution algorithms will allow very large mutation changes to occur if there are two different population members. The selection operator operates by selecting individuals that are different to the current individuals observed so far this is done by mutating existing members of the population. [30][72]

The hyperparameter optimisation process for the DEC model described above optimises for: the DEC model’s densely connected layers, the amount of nodes in those densely connected layers, the amount of dropouts in the DEC model, the dropout layer weights and the amount of clusters in the clustering layer of the DEC model. The genetic algorithms optimise the DEC model for a minimal silhouette evaluation metric, which does not require ground truth labels for the data, being calculated using the clusters and the data. Higher scores for the silhouette evaluation metric mean that clusters are dense and well separated. [12] As mentioned in chapter 2 section 2.3, the genetic algorithms were implemented from scratch, the DEC model was implemented using the Keras deep learning library and the silhouette and other evaluation metrics were calculated by using the implementations provided in the sci-kit learn library. The genetic algorithms process generates the code for the DEC model in a lazy fashion and then executes that code, meaning that the implementation of the DEC model makes use of template code which is populated using the hyperparameters computed by the genetic algorithms process. Invalid solutions that are encountered, such as a DEC model with a dense layer with no nodes, will result in a dense layer with 1 nodes. Such solutions are unlikely to be encountered, but could occur upon initialisation of the genetic algorithms. Other edge cases such as this are handled in a similar fashion, which is to say any invalid solution will be calculated to be the nearest valid solution.

4.5 Results & Discussion

The DEC model that has undergone hyperparameter optimisation by way of genetic algorithms achieves a silhouette value of approximately 0.37 for the CIFAR-10 dataset and 0.23 for the Fashion-MNIST dataset. The score for the CIFAR-10 dataset indicates that there is a weak structure within the clusters, but a structure all the same. The score for the Fashion-MNIST dataset indicates that no substantial structure can be found within the data. [73]

Further investigation into the clusters was conducted, revealing that 5,999 of the instances were in the one cluster for the CIFAR-10 dataset and 6,997 for the Fashion-MNIST dataset.

Label	Amount of images
Misc	5,999
Cat	1

Table 4.5.1: Labels determined in the CIFAR-10 dataset by the clustering framework

Table 4.5.1 shows the labels that were assigned to the CIFAR-10 dataset (using the test split defined by the Keras framework) by the DEC model that underwent hyperparameter optimisation using genetic algorithms (maximising the silhouette evaluation metric). We can also see in table 4.5.1 the amount of images assigned to each label. The label names are defined by the CIFAR-10 dataset, with the exception of misc. The misc label is defined by the author as there is no pattern among the images assigned this label, as nearly all the images have been put together. This means the accuracy of the clustering sub-framework on the CIFAR-10 dataset is 0.1000167%, an extremely low score considering that an accuracy of 70% is what is considered a reasonable accuracy value, or in comparison to using the DEC model without hyperparameter optimisation, which achieves a 30% accuracy value. A baseline model that randomly assigned a label from 10 possible labels to the images for the CIFAR-10 test dataset (10 being chosen as this is the number of categories in the dataset), which achieved a slightly higher accuracy value of 0.1033%. In other words, it is actually better to randomly select labels for the data than to use the clustering sub-framework developed here. Thus, the clustering sub-framework developed for this dissertation was shown to be ineffective on the CIFAR-10 dataset.

Misc	6,997
Coat	2
Pullover	1

Table 4.5.2: Labels determined in the Fashion-MNIST dataset by the clustering sub-framework

Table 4.5.2 shows the labels that were assigned to the Fashion-MNIST dataset by the DEC model that underwent hyperparameter optimisation using genetic algorithms (maximising the silhouette evaluation metric). We can also see the amount of images assigned to each label. The label names are defined by the Fashion-MNIST dataset, with the exception of the misc. The misc label is defined by the author, as there is no pattern among the images assigned this label (with nearly all images having been assigned it). The accuracy of the clustering sub-framework on the Fashion-MNIST dataset is 0.1000167%, which is extremely low. As discussed already, typically an accuracy value of 70% is what is considered reasonable, and even the DEC model without hyperparameter optimisation achieves a 30% accuracy value on the CIFAR-10 dataset. A baseline model that randomly assigned a label from 10 possible labels to the images for the Fashion-MNIST test split (10 being chosen as this is the number of categories in the dataset) achieved a slightly higher accuracy value of 0.1043%. Once again, random label assignment actually outperformed the sub-framework developed here with the difference between the two not being statistically significant. Therefore, the developed clustering sub-framework was shown to be ineffective on the Fashion-MNIST dataset.

For the most part, on both datasets, the images were just being clustered into one cluster (with a few images being clustered into others). This method did not even outperform the random baseline models, with respect to accuracy. Therefore, we can conclude that using genetic algorithms to perform hyperparameter optimisation on the DEC model (maximising the silhouette evaluation metric) was unsuccessful. The ineffectiveness of this approach could be due to the high dimensions of the data, or the silhouette metric not having been appropriate as an optimisation metric.

4.6 Conclusion & Future Work

This method of clustering images was found to be largely ineffective, clustering most of the data into the same category, and having accuracy that was lower than the state-of-the-art supervised methods, the DEC model without having undergone hyperparameter optimisation (maximising the silhouette evaluation metric) and even the random baseline model. This means that the answer to the research question that this chapter addresses, "*Is the Ogma framework able to effectively label images using their contents?*", is no. While the Ogma framework would be able to label the CIFAR-10 and Fashion-MNIST datasets effectively if the state-of-the-art methods were used, this would make the operation of the Ogma framework more of a manual, which would defeat its purpose. However, for the Ogma framework to be able to effectively label images using their contents, an accuracy of value at least 70% from the clustering sub-framework would need to be achieved. Furthermore, it is clear that optimising the model with respect to minimising the silhouette metric was not useful. Another reason this method likely did not work is due to the high dimensions of image data.

This method could be improved by performing dimensionality reduction on the image data before providing it to the sub-framework. The silhouette metric could be made more suited to image data by using a convolved version of the metric, which would apply the convolution operation to images in the input, or perhaps applying max-pooling or mean pooling to the images. Another way in which this method could be improved would be through the invention of a new evaluation metric based on the silhouette evaluation metric, which would analyse clusters using sections of pixels as blocks to reduce the dimensionality, taking a 2x2 window and treating it as its own entity. A potential issue with this would be the rotated images of the same category would be put into different clusters, as this window would be subject to the rotation of the image if the dimensions of the image were odd. Due to time and computational constraints, these methods were not explored.

In summary, applying the DEC model with hyperparameter optimisation using genetic algorithms (maximising the silhouette evaluation metric) has been shown to be largely ineffective for the Fashion-MNIST and CIFAR-10 datasets. Due to this finding, this sub-framework was not used in the search functionality of the ontology, as the majority of the data was put into one cluster. If this sub-framework had been found to be effective, then the same method could also have been applied to image segments, allowing for search functionality to be introduced to the ontology whereby an image segment could be uploaded and similar image segments or images containing similar image segments would be returned.

5 Colour Extraction Ogma Sub-Framework

5.1 Overview

This chapter will discuss the colour extraction sub-framework of the Ogma framework.

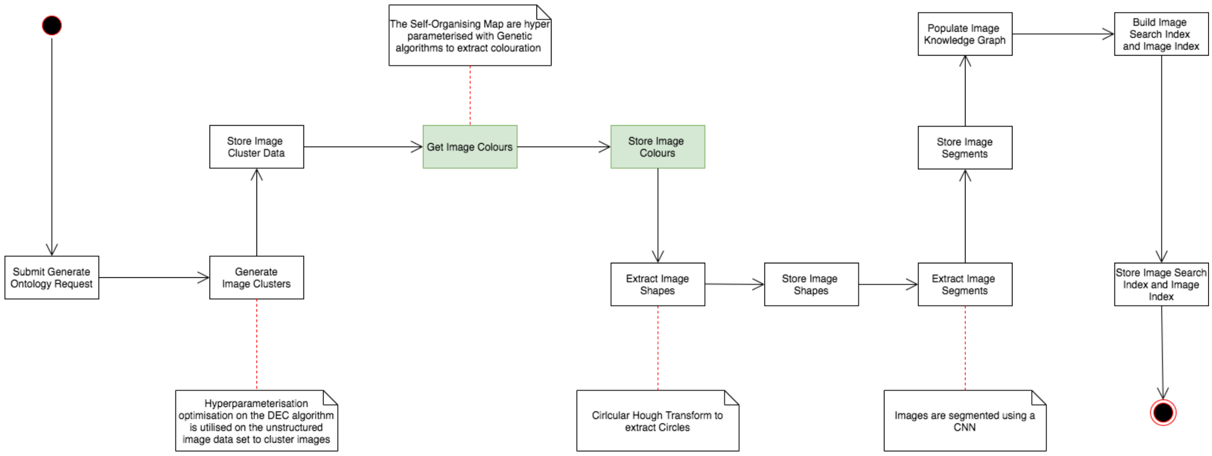


Figure 5.1.1: Activity diagram representing the Ogma framework with the Colour extraction processes highlighted

To contextualise the components of the Ogma framework that are discussed in this chapter the colour extraction sub-framework in the Ogma framework are highlighted as these are discussed in this chapter. The research question which this chapter addresses is *"Can the Ogma framework effectively extract dominant colours from images?"*.

This colour extraction sub-framework sub-framework is responsible for the extraction of colours from the images when they are inserted into the ontology, extracting between 1 and 8 colours from each image (1 where the image is all the same colour, up to a maximum of 8 where there are multiple colours this is performed using genetic algorithms as the hyperparameter optimisation technique). The extracted colours are stored in hexadecimal format and are also generalised (by way of clustering) to the nearest colour in the CSS3 colour standard based on Euclidean distance.

Figure 5.1.2 Example image of a cat from the CIFAR-10 dataset

The reason why the colour extraction sub-framework extracts only the dominant colours and not all of the colours within an image is because images often tend to actually contain a vast number of different colours, even if only a handful of these colours make up significant portions of the image. For example, the image of a cat shown in figure 5.1.2 contains a total of 960 unique colours (on the RGB colour space), even though only about 6-8 colours would appear to be really important in the image Extracting all of these colours from every image would significantly impair the search functionality of the Ence knowledge graph, as searching for a particular colour in this case would return every image in which even one pixel corresponds to this colour.

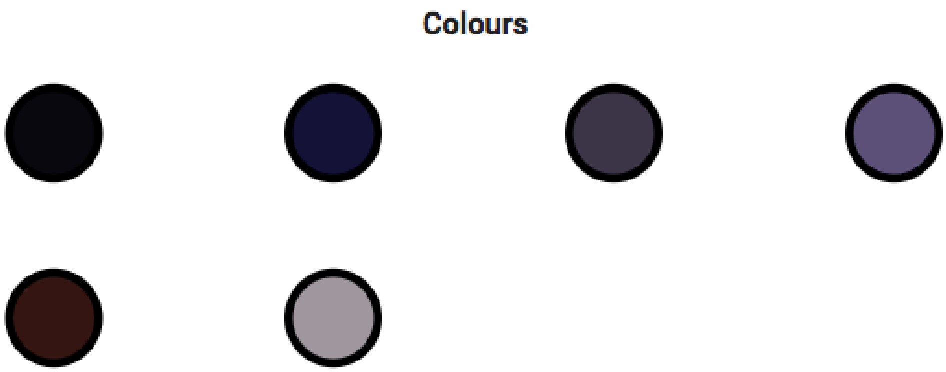


Figure 5.1.3: Colours extracted from figure 5.1.2

Figure 5.1.3 shows the colours that were extracted from figure 5.1.2 by the colour extraction sub-framework. We can see from looking at the image that these are its most dominant colours (such as the dark red background). This example would appear to justify the design choice taken to limit the number of dominant colours to 8, as in this case it would appear that no colours that are important in the images have been left out.

5.2 Background

There has been little research into extracting a variable number of colours (in this case between 1 and 8) from images as is done by the framework discussed in this chapter. The closest field of research to the methods discussed in the chapter is that of colour quantisation, which aims to reduce the number of colours in images while minimising the amount of visual distortion. Colour quantisation was originally used for the visual displays of devices which could only render a limited number of colours.The state of the art of colour quantisation is discussed in this section in order to determine the algorithm that should be used in the development of this sub-framework.

In the field of colour quantisation, the size of the colour palette is typically between 8 and 256, making such methods appropriate for the application discussed in this chapter (where the colour palette size is 8). An evaluation metric that has been shown to be highly widely used in colour quantisation is the SSI. [76] Deep learning techniques utilising convolutional neural networks are not implemented, as these would be too computationally intensive to use with the genetic algorithms process for optimising the number of colours selected (the colour quantisation process is run on each image in both datasets).

One of the most notable early methods of colour quantisation was the median cut algorithm. This algorithm sorts data of some number of dimensions into sets which contain vectors of the data, then recursively cuts each of these sets at the median point along its longest dimension. [77] [78]
More recent approaches to colour quantisation involve using clustering methods, the most notable of which are the fuzzy c-means, the k-means and self organising maps (SOM), each of which have been shown to be effective, with SOM specifically having been shown to be particularly effective. SOM is a type of artificial neural network which uses competitive learning as the training algorithm instead of backpropagation. [11][79] Competitive learning is a learning strategy where the nodes in the neural network are in competition with one another to activate in response to input data. [81] For background information on the approach to genetic algorithms that will be used in this implementation, please see chapter 3, where genetic algorithms are discussed with regard to clustering. The names of the colours are calculated by mapping the hexadecimal RGB colour value of a pixel to the nearest colour in the CSS3 standard (measured by Euclidean colour space, which is widely used for calculating the differences between colours) and assigning it the name of attached to that colour in the CSS3 standard. [82]

5.3 Implementation

The implementation of the colour extraction sub-framework used the MiniSOM library with genetic algorithms as the hyperparameter optimisation method that were implemented from scratch. The genetic algorithms process was run for 30 generations, with 100 members, such that each new generation would replace all the members in a previous generation. The selected crossover rate and mutation rates were 0.5 and 0.05, respectively. These rates were chosen as these are commonly chosen effective values in genetic algorithms for the crossover and mutation rates [72]. Tournament selection was used in the genetic algorithms selection process, with 7 random individuals being selected from each population and the fittest being returned as the sample of that population. The reason why tournament selection was used is to keep a degree of fresh individuals coming into the population and getting rid of the old population members as each of the SOM individuals execute relatively quickly there is no significant computational to performing tournament selection. [72] The genes of each individual are encoded using fixed length arrays composed of 1s and 0s [30]. The SOM model was evaluated using the silhouette distance metric of the clusters plus the SSI metric as this was found to perform better in the tuning of the fitness function of the genetic algorithms, with a higher value indicating clusters of colours that are better separated in the RGB colour space.

The SOM model parameters that undergo hyperparameter optimisation are the model's number of clusters, its sigma value, its topology and its neighbourhood. The sigma value is the distance that the neighbourhood function covers in the covers with respect to the dimensions to the map of the SOM model. The topology of the SOM model is how the nodes in the artificial neural network are connected, in this case the topology can be rectangular or hexagonal. The neighbourhood function is a function which assigns weights to positions in the SOM model. The values for this are gaussian, bubble, triangle and mexican hat. [83] As the model uses a colour space of 8 colours, the maximum number of colours the model can use is 8. The model will return the colour of the clusters in hexadecimal format and will use name colours on the basis of their closest colour in the CSS3 colour scheme, allowing for named colour searches to be carried out by the search engine, as discussed in chapter 4. To extract colours from images in parallel the GNU parallel library was used. [24] The web colours library was used to get the taxonomy of CSS3 colours and the euclidean distance function (implemented from scratch and integrated with the web colours library) was used with this to determine the closest colour in the CSS3 library if it did not exist.

5.4 Results & Discussion

The SOM model that has undergone hyperparameter optimisation by way of genetic algorithms achieves a mean (every image is evaluated individually) silhouette and SSI value of approximately 0.78 and 0.93, respectively, on the Fashion-MNIST and 0.84 and 0.49 on the CIFAR-10 dataset. The SSI values are calculated by creating an image in which only the extracted colours are used, with this image generated from the clustered data being compared against the original image to compute the SSI values. The silhouette evaluation metric of 0.78 for the Fashion-MNIST has a strong cluster structure, meaning that the colour clusters for the Fashion-MNIST dataset are dense and well separated, and can hence be considered to be well clustered. The SOM models mean SSI value of 0.93 on the Fashion-MNIST dataset indicates that the colours that are extracted from the images are highly representative of the colours in the original image, with the maximum SSI value being 1 (with 1 indicating no structural difference between the two images and 0 indicating no structural similarity). Higher values of the SSI evaluation metric indicate that there is a strong structural similarity between the images which is the case with the Fashion-MNIST dataset thus the colour extraction sub-framework is performing effectively on the Fashion-MNIST dataset. [84][85][87] Therefore, it can be said that the colour extraction sub-framework performs effectively on the Fashion-MNIST dataset. The mean silhouette evaluation metric of 0.49 achieved on the CIFAR-10 dataset indicates a weak cluster structure, meaning the colour clusters are someway dense and someway well separated. On the other hand, a reasonably strong SSI metric of 0.84 was achieved. However, this was still lower than that which had been achieved for the Fashion-MNIST dataset, and overall it can be concluded that the SOM models are not as effective at extracting colour data from the CIFAR-10 dataset as from the fashion-MNIST dataset, possibly due to more information being contained in the images of the CIFAR-10 dataset.

A k-means baseline model that extracted colours into 3 clusters (meaning that a colour palette of 3 colours would be extracted for every image) was created to compare against the proposed colour extraction sub-framework. The reason why 3 was chosen is that the images are being analysed in the RGB colour space (meaning 1 cluster potentially for each channel in the baseline model). The k-means baseline model mean (every image is evaluated individually) silhouette and SSI values were approximately 78.1 and 0.566 (respectively) on the CIFAR-10 dataset and 0.89 and 0.81 on the Fashion-MNIST dataset. We can see that the k-means baseline model performs reasonably well in extracting the colours from both datasets, both in terms of its silhouette and SSI values. The k-means baseline model created denser clusters and more well separated clusters, which we can see from the silhouette evaluation metric. However, this was at the cost of the SSI evaluation metric, which is more important in the context of colour extraction, as this ensures that colours are being effectively extracted. Therefore, it can be concluded that the proposed colour clustering framework outperforms the k-means baseline model reasonably well.

5.5 Conclusion & Future Work

In conclusion, the use of genetic algorithms to perform hyperparameter optimisation for the SOM algorithm was shown to be largely successful, as the SOM models were able to effectively extract colours from the CIFAR-10 and Fashion-MNIST datasets, judging from the mean SSI values achieved. Therefore, the research question "*Can the Ogma framework effectively extract dominant colours from images?*" can be answered with yes.

This sub-framework could be improved by further parallelising the genetic algorithms hyperparameter optimisation process, allowing for more generations to be completed within the same time constraints. The colour extraction sub-framework could also be extended by exploring the use of convolutional neural networks instead of the SOM algorithm. Another way in which the colour extraction framework could be extended is by extracting colours from non-still image data such as GIFs or video data.

6 Image Segmentation Ogma Sub-Framework

6.1 Overview

This chapter outlines the Image Segmentation sub-framework of the Ogma framework, which segments images for insertion into the Ence ontology. This framework segments images in an unsupervised fashion, utilising the current state of the art of unsupervised image segmentation methods. [32]

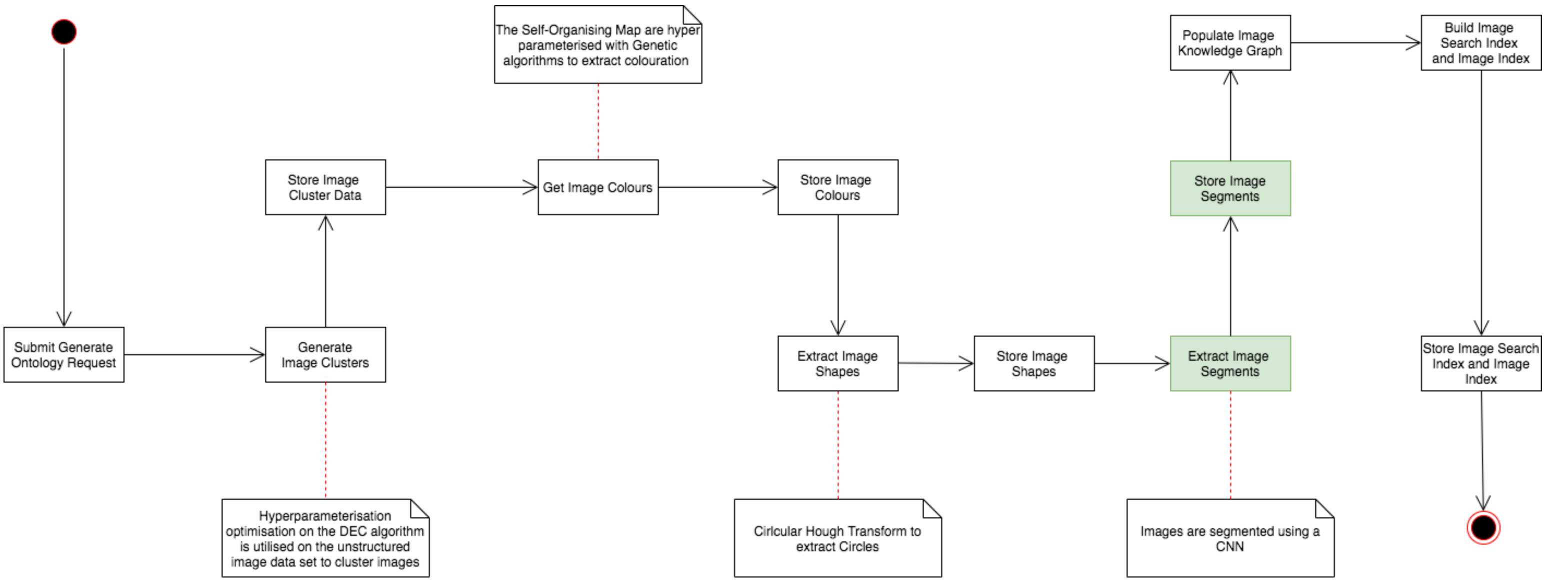


Figure 6.1.1: Activity diagram representing the Ogma framework with the Image Segmentation processes highlighted

The research question that is addressed in this chapter is "*Can the Ogma framework effectively segment images in an unsupervised manner?*".

6.2 Background

Image segmentation is the task of splitting an image into meaningful regions which logically belong together. [8] This task takes into account the position of a pixel in an image and how the pixel relates to other pixels in the image. [8] The most commonly used evaluation metric for object detection and image segmentation is the intersection over union (also known as the Jaccard index) evaluation metric. The intersection over union evaluation metric measures the overlapping area between a ground truth for a segment and the area predicted for segmentation divided by the total area of union and is computed as a ratio. Essentially, the intersection over union evaluation metric for a given segmentation problem is the measure of how closely the area of the predicted segment matches with the defined segment. An earlier method used to segment images was region splitting, which used the phagocyte algorithm, a divisive algorithm. [88] More recent algorithms for image segmentation optimise some loss function which is subject to the specific algorithm, such as the intra-region boundary lengths and intra-region consistency or dissimilarity algorithms. [8] Clustering is also a widely used technique in the area of image segmentation (in which this approach is referred to as mean-shift and mode finding methods), including popular clustering algorithms such as k-means. The mean-shift and mode finding methods model each pixel with feature vectors (the colour of the pixel and the position for example) and treat these feature vectors as a sample from a probability density function and assign pixels to clusters in the probability density function. Deep learning techniques have also been applied, including artificial neural networks and convolutional neural networks. [8][32]

6.3 State of the Art

There is quite a lot of research into semi-supervised and supervised image segmentation methods, but not for fully unsupervised image segmentation methods. There is also a lack of research into the CIFAR-10 and Fashion-MNIST datasets. As these datasets do not have ground truths for the image segments, the best performing model in this section is implemented based on the mean over intersection score computed on the PASCAL VOC 2012 and the BSD500 datasets, which are commonly used image segmentation datasets. [32] The state of the art methods discussed in this section are unsupervised but are evaluated using the PASCAL VOC 2012 and the BSD500 gold standards to determine if these unsupervised methods are performing effectively. The identified state of the art method that will be primarily focused on in this section the are the model architectures discussed in the paper "Unsupervised Learning of Image Segmentation Based on Differentiable Feature Clustering".[32]

The current state of the art unsupervised method for use on the PASCAL VOC 2012 dataset, having regard to the mean intersection over union evaluation metric, are the model architectures discussed in the paper "Unsupervised Learning of Image Segmentation Based on Differentiable Feature Clustering". The graph cut model in this paper achieved mean intersection over union values of 0.3647 and 0.3078 on the VOC 2012 and BSD500 respectively, while the method proposed by this paper achieved mean over intersection values of 0.352 and 0.3225 on the VOC2012 and BSD500 datasets respectively. Therefore, the state of the art approach outperforms the approach to segmentation put forward in this paper with respect to the intersection over union metric when applied to the VOC2012 dataset, although not when applied to the BSD500 dataset. However, the degree to which the method proposed in this paper outperforms on the BSD500 dataset is higher than the extent to which it is outperformed on the VOC2012 dataset (as measured by intersection over union), and hence the method proposed in this paper was still used in the image segmentation sub-framework. [32] As there is a severe lack of research that using fully unsupervised segmentation deep learning techniques, graph cuts are compared to the state of the art method discussed here. [86]

This paper's proposed method uses a convolutional neural network with differentiable feature clustering to perform unsupervised image segmentation. The method operates by generating a satellite image based on the number of pixels in the image, these pixels then being normalised to be within the values 0 and 1. Feature extraction is then performed with a mapping function of the feature vector created from the normalised pixels. The pixels are then assigned into clusters labels that are calculated to be unknown are predicted by using the previously mentioned feature extraction and mapping function, after the initial features are extracted and mapped to clusters by way of the mapping function. The mean intersection values that are achieved by these are somehow effective. The convention is that 0.5 is an effective value for segmenting images but we can see in the results section of this chapter that a lower (and hence better) score is achieved.

6.4 Implementation

The implementation of the image segmentation sub-framework utilises the methods discussed in the Unsupervised Learning of Image Segmentation Based on Differentiable Feature Clustering paper. The work presented in this paper was not expanded upon as the CIFAR-10 and Fashion-MNIST datasets are do not have ground truths for the segments of their images, and it would not be feasible due to time constraints to create these with the CIFAR-10 and Fashion-MNIST datasets have 60,000 and 70,000 images in total, respectively. [3][4][32]

The image segmentation sub-framework was implemented using the python programming language with the PyTorch deep learning library. The implementation also uses the OpenCV library in order to load the images into the sub-framework in order to segment them. The option of passing in scribbles discussed in the Unsupervised Learning of Image Segmentation Based on Differentiable Feature Clustering paper was not implemented as this is not relevant to this dissertation. The hyperparameters for the Convolutional Neural Network and other configurations remained the same in this implementation as in the implementation proposed in the Unsupervised Learning of Image Segmentation Based on Differentiable Feature Clustering paper.

6.5 Results & Discussion

As the mean intersection over union could not be calculated due to the lack of existence of a segmentation gold standard for the CIFAR-10 and the Fashion-MNIST datasets, a random sample of 100 segmented images each from the CIFAR-10 and Fashion-MNIST images datasets were evaluated.

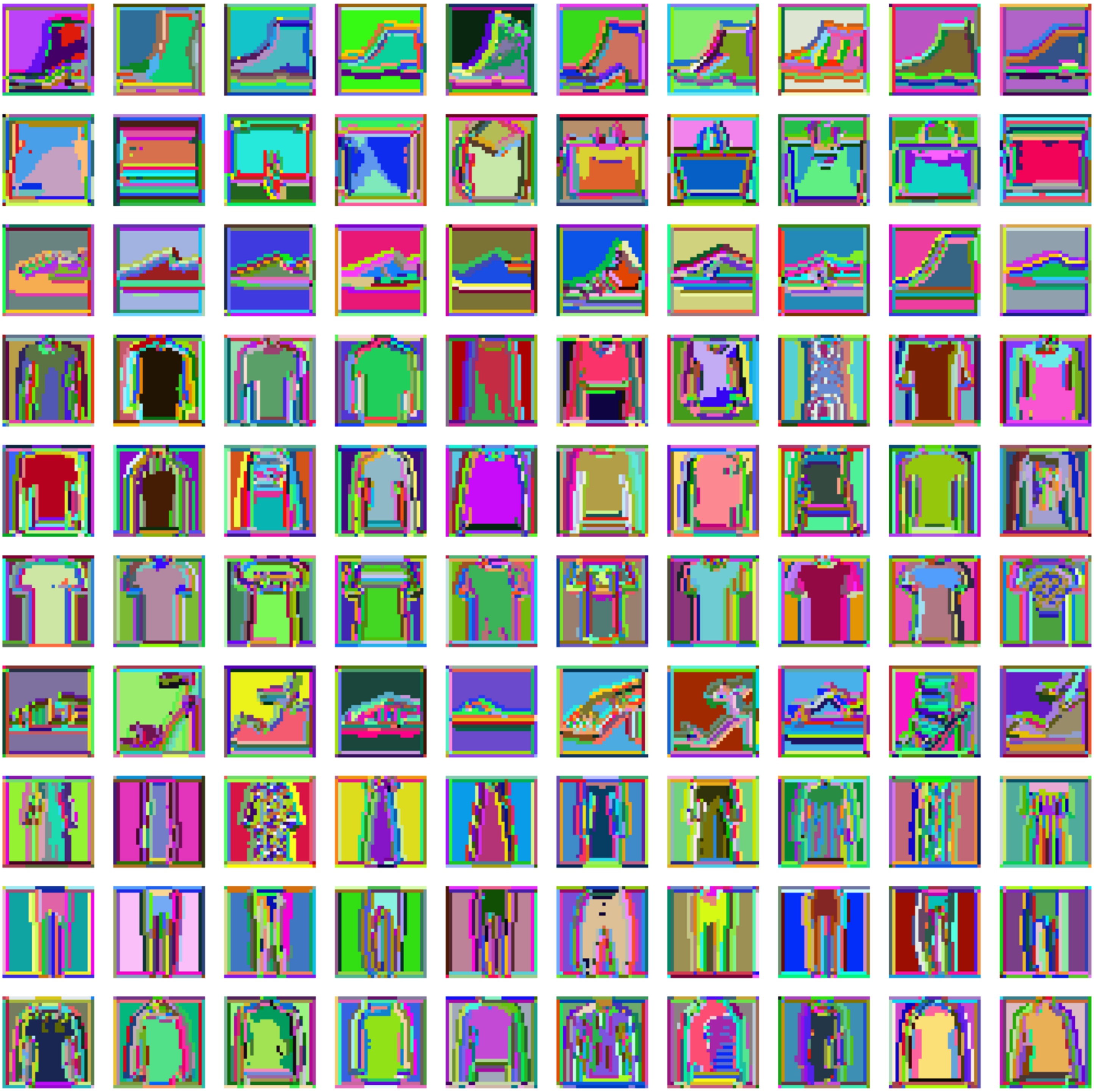


Figure 6.1.2: Fashion-MNIST Evaluated Segmented Images

Figure 6.1.2 shows a 10x10 facet grid of the 100 images sampled from each of the 10 categories in the Fashion-MNIST dataset. In the segments, each colour represents a unique segment of the image. A segment can be thought of as a component of an image, for example, we can see that for the image in row 2 column 7 the background, bag body, bag edges and bag handle all have their own segment in the image. We can see from figure 6.1.2 that the images appear to be logically segmented for the most part. For example, we can see that the boots on the top row all appear to be split up, highlighting the different components of the boot (e.g. side, heel). We can see that the background of the images is also being treated as its own segment, which is good as the background is not being blended with the item itself. One image that has not been segmented well is the image at row 1 column 1, which appears to have quite a lot of noise. However, for the most part, the images are well segmented, especially the image at row 7 column 9. However, the source images do appear to have quite a bit of noise, which is why there are a lot of segments. A potential solution to this could be to use denoising techniques on the data before passing the data to the segmentation sub-framework.

Figure 6.1.3: CIFAR-10 Evaluated Segmented Images

Figure 6.1.3 shows a 10x10 facet grid of the 100 images sampled from each of the 10 categories in the CIFAR-10 dataset. These images would appear to be, for the most part, logically segmented. There are a lot more segments in these images compared to the Fashion-MNIST dataset, which makes sense considering the fact that the source images are more detailed. For example, rows 2 and 3 contain images of cars and boats, respectively, with the different parts of these vehicles being segmented (e.g. body vs. wheels for cars). However, another explanation for this may be that this segmentation method has more difficulty generalising the CIFAR-10 dataset than the Fashion-MNIST dataset. Two of the most effectively segmented images in figure 6.1.3 would appear to be the truck at row 7 column 3 and plane at column 6 column 10. We can see here that the background of the images has also been segmented into different components, unlike for the Fashion-MNIST dataset where the background has been given a single segment. This is likely due to the background for the Fashion-MNIST images being a single colour, while for the CIFAR-10 dataset many of the images are outdoors and hence the background may be composed of sky, grass, etc., for example in the image at row 10 column 3. Over all, the CIFAR-10 appears to be segmented reasonably well, as all of the unique parts of the images are given their own segments.

6.6 Conclusion & Future Work

The methods adopted for use in the image segmentation sub-framework of the Ogma framework have been manually evaluated as reasonably successful when applied to the data samples collected from the Fashion-MNIST and CIFAR-10 datasets. Given this, we can say that the research question addressed in this chapter "*Can the Ogma framework effectively segment images in an unsupervised manner?*" can be answered with a yes, as the image segmentation sub-framework is able to segment the images based on their contents. However further research could entail denoising the input images to reduce the number of segments created.

The work presented in this chapter could be further expanded by developing a gold standard for the CIFAR-10 and Fashion-MNIST datasets, which would allow for further tuning of the CNN model utilised by having a reference point to verify if images are being better segmented. Another way in which the image segmentation sub-framework could be expanded on would be if a hyperparameter optimisation method such as genetic algorithms were to be applied in the tuning of the CNN segmentation model, optimising for mean intersection over union or some other evaluation metric. Other future work might include performing hyperparameter optimisation on the VOC2012 dataset, with the aim of maximising mean intersection over union.

7 Shape Extraction Ogma Sub-Framework

7.1 Overview

This chapter discusses the shape extraction sub-framework of the Ogma framework. This framework is responsible for extracting shapes from the images that are provided to the Ogma framework. Only circles are extracted from the set of images as it was determined that circles were the only shapes which could be accurately extracted from the set of images in an unsupervised manner.

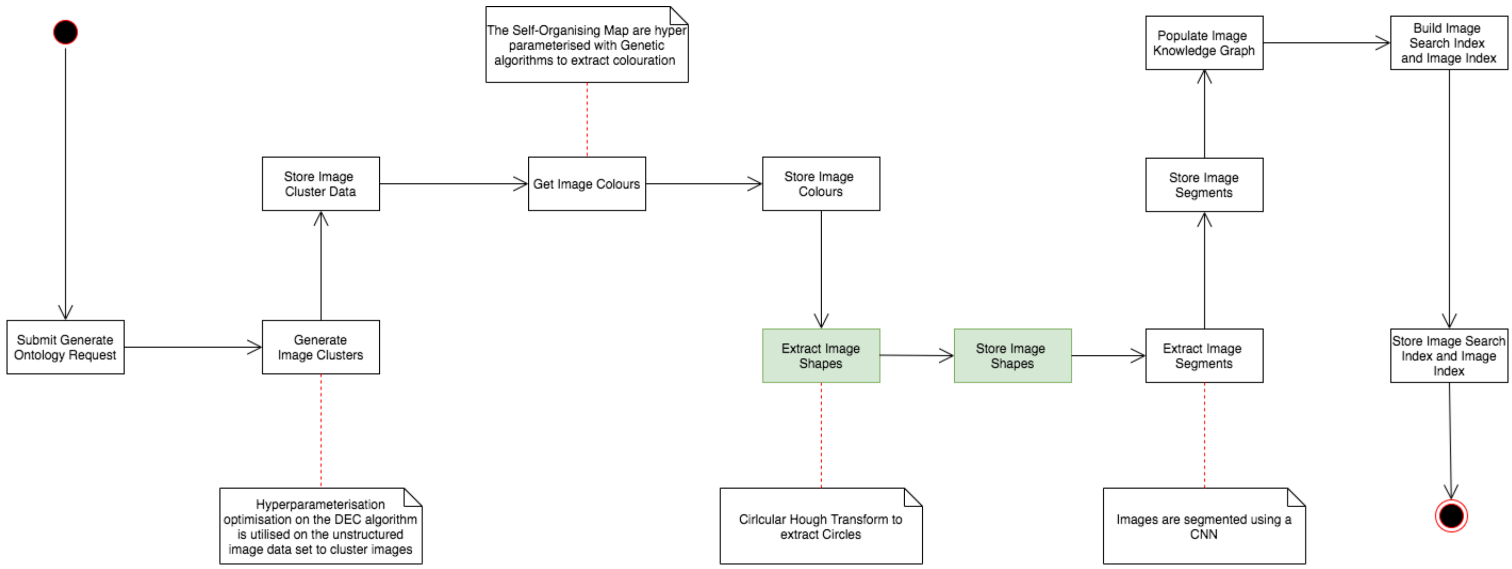


Figure 7.1.1: Activity diagram representing the Ogma framework with the Shape Extraction processes highlighted

To contextualise the components of the Ogma framework that are discussed in this chapter the shape extraction processes in the Ogma framework are highlighted as these are discussed in this chapter. The research question that is addressed in this chapter is "*Can the Ogma framework effectively extract shapes in images in an unsupervised manner?*".

The sub-framework described in this chapter will apply state-of-the-art unsupervised shape extraction techniques to still image datasets.

7.2 Background

Detecting shapes within images has been performed using techniques such as contour detection, however, this method is unsuitable for the shape extraction sub-framework as while you will be able to extract shapes from images it is very difficult to quantify what those shapes actually are using contour based extraction. [8] [89] Template matching is another common shape extraction approach, however the issue with template matching is that a template matching techniques are sensitive to rotation and scale meaning they will not extract images where the template differs, and template matching techniques do not generalise well enough to just provide an image of a shape and to extract that particular shape from images which would be required in this case thus making template matching techniques not suitable for the shape extraction sub-framework. State of the art shape extraction methods are highly effective but use supervised learning in order to extract the shapes from images thus making these methods unsuitable for the shape extraction sub-framework. [90] [91] [92] There appears to be a significant lack of research in unsupervised shape extraction from images with regards to shapes like triangles, squares and pentagons etc. however there is significant research into extracting circles from images with the CHT algorithm, thus these limitations are what defined the scope of the shape extraction sub-framework.

The CHT algorithm is a highly effective way of extracting circles from images in an unsupervised manner and was thus used in the shape extraction sub-framework. The CHT is used to detect circles in an image by searching for objects that have highly symmetrical radial degrees. [17]

7.3 Implementation

The shape extraction sub-framework of the Ogma framework uses the OpenCV library in implementing the CHT algorithm. This implementation of the CHT algorithm makes use of the Canny operator to obtain the binary edge points within images (as defined in A New Modified Hough Transform Method for Circle Detection) [93]. To automatically select the hyperparameters of the Canny shape extraction kernel in the CHT algorithm, the Otsu method is followed using the OpenCV library to compute the high and low thresholds for the CHT algorithm. [34] The CHT algorithm parameters have been tuned with the aim of extracting circles that are significant to the image, with circles below a certain relative size being ignored as noise. The minimum radius of a circle contained within the image must be 10 percent of either the image’s height or width (whichever is the smaller) and the minimum distance that must exist between circles centers is 3 percent of the image's height or width (again, whichever is smallest).

7.4 Results & Discussion

The CHT is a circle extraction method which has no shortage of literature supporting its effectiveness. However, to evaluate this method's validity for use in this application, a sample of the circles extracted by the method will be evaluated. Only a sample is taken as there are 70,000 images and 60,000 images in the Fashion-MNIST and CIFAR-10 datasets, respectively, making it infeasible to evaluate every single image. [93] No circles of the specified dimensions were extracted from the Fashion-MNIST dataset, thus the circles from the CIFAR-10 dataset will be evaluated.

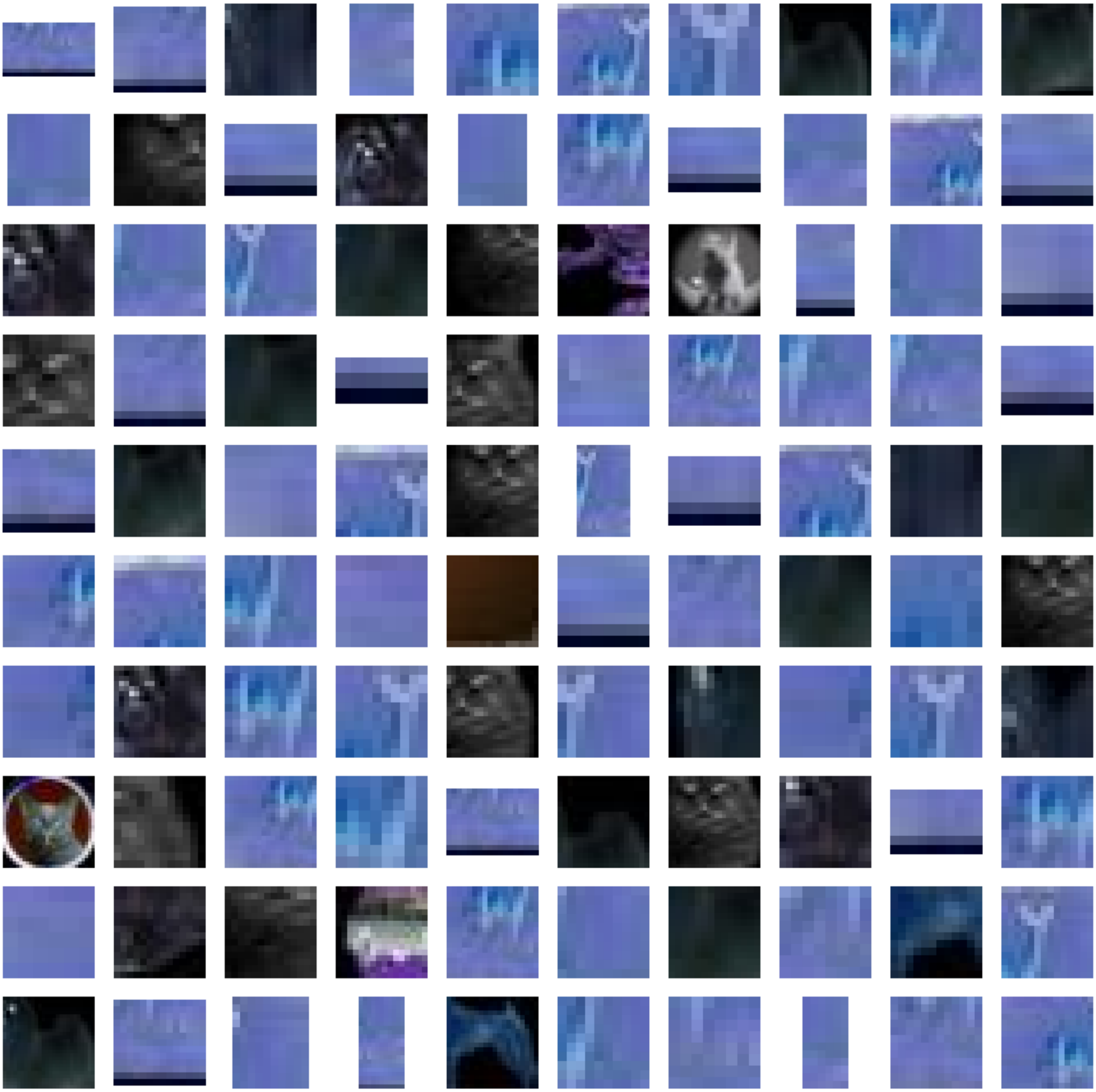


Figure 7.1.2: Evaluated Images which circles were extracted from

Figure 7.1.2 shows a 10x10 facet grid of the 100 images evaluated of the total 173 images which circles which circles were found to be present within using the CHT algorithm. We can see immediately from the set of images some obvious examples which circles are present within the images these being the image on the second row on the 4 column showing the dogs head as a circle extracted the same with some of the other examples such as the body on a car, the circle surrounding a cats face, circle surrounding a dogs face. One of the less obvious cases would be the deer's head in the 3rd row and the 3rd column of the facet grid which shows a deer's head.

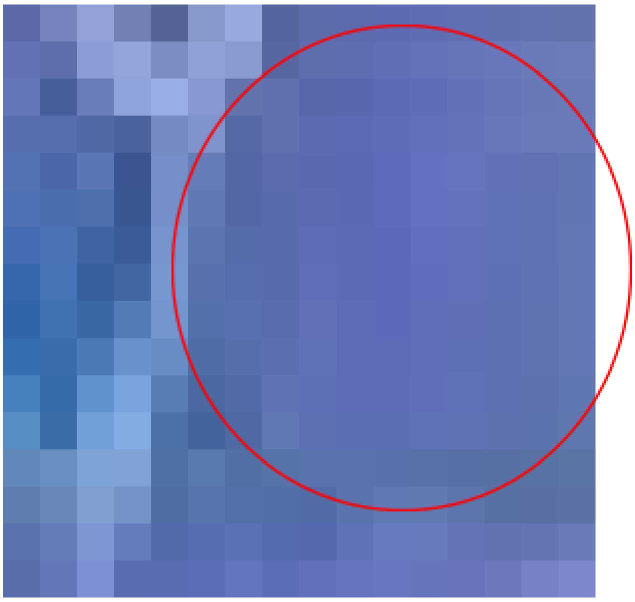


Figure 7.1.3: Deers head in row 3 column 3 of the facet grid shown in figure x

Figure 7.1.3 shows the deer's head from the facet grid shown in figure 7.1.2 in the 3rd row and the 3rd column of the facet grid which shows a deer’s head. Upon closer inspection we can see that the background appears to have a darker blue circle inside a lighter blue circle when you look closely at the image which is not apparent from initial inspection.

The method of using a hough transform to extract circles from the images in an unsupervised manner has been successful in that any image in which a circle has been detected a circle does exist interested readers can find the full set of images in which circles have been extracted from in appendix C in the folder circle, the binary images can be found in the "binary_images" folder in addition to this where one can further see why the hough transform has listed images to have circles within them.

7.5 Conclusion & Future Work

In conclusion the circle extraction sub-framework uses the CHT algorithm which is a very mature method for extracting circles from images we can see the circle extraction sub-framework discussed in this chapter has been shown to be successful in that it is able to effectively extract circles from images from the manual evaluation that was conducted. This means that the answer to the research addresses “*Can the Ogma framework effectively extract shapes in images in an unsupervised manner?*” is yes, the Ogma framework is able to effectively extract circular shapes from images.

The shape extraction sub-framework could be further extended upon by utilising Markov Random Fields to attempt to extract other shapes within images such as squares, triangles or polygons etc. Another method in which the shape extraction sub-framework could be further extended upon is by utilising denoising techniques such as autoencoders to prevent background information being detected as a shape to attempt to only have shapes that are significant to the image be detected and to allow for smaller circles to be extracted.

8 Conclusion

Ence Ontology

The creation of the Ence ontology model to represent the contents of unstructured, still image data (extracted by the Ogma framework) is concluded to be essentially successful, although only around half of the competency questions could be successfully answered in this implementation. More than half of the competency questions posed in chapter 3 were successfully answered, however due to the ineffectiveness of the Ogma framework's image clustering sub-framework (discussed in more detail chapter 3), the competency questions which relied on the data that was to be extracted using this sub-framework could not be successfully answered (those being questions 1, 5, 6 and 7).

The primary research question that this dissertation has posed, which was "*To what extent does the Ence ontology effectively model data generated by the Ogma framework from an unstructured set of still images?*" can however be answered in the affirmative, as the Ence ontology was able to effectively model the data generated by the Ogma framework - the main issue being that the clustering sub-framework of the Ogma framework did not function as intended. In other words, the failure of the Ence ontology to successfully answer competency questions 1, 5, 6 and 7 was an issue with the data it received rather than the structure of the ontology itself. However, as the data for image shapes, segments and colours are effectively extracted by the Ogma framework, competency questions 2 - 4 and 8 - 10 are effectively answered using the CIFAR-10 and Fashion-MNIST datasets.

Ogma Framework

The development of the Ogma framework was successful with respect to the extraction of colours, segments and circles from unstructured sets of still images (the image datasets used to verify this were the CIFAR-10 and Fashion-MNIST datasets). The Ogma framework, however, was unable to effectively cluster unstructured sets of still images based on their contents alone.

Image Clustering Ogma Sub-Framework

The image clustering sub-framework of the Ogma framework was found to be largely ineffective as it clustered most of the data into the same category, having accuracy that was no better than using the DEC model without hyperparameter optimisation methods or the random baseline model. Therefore, it could not be said to outperform the existing state-of-the-art methods. The DEC state-of-the-art model which had undergone hyperparameter optimisation using genetic algorithms (with the goal of maximising the silhouette evaluation metric) did not outperform either the state-of-the-art supervised method or the DEC model without hyperparameter optimisation in terms of accuracy. Hence, the development of the proposed clustering sub-framework outlined in this dissertation is deemed to be unsuccessful.

As the image clustering sub-framework was found to be ineffective, the answer to the research question addressed by this sub-framework, *"Is the Ogma framework able to effectively label images using their contents?",* is no. The reason for this failure was likely due to the high dimensional of the data, which appeared to cause issues with optimising for the silhouette evaluation metric when applying hyperparameter optimisation to the DEC model.

Colour Extraction Sub-Framework

The colour extraction sub-framework was found to be effective at extracting colours from images, meaning the answer to the research question that this sub-framework addresses, "*Can the Ogma framework effectively segment images in an unsupervised manner?",* is yes. Performing hyperparameter optimisation using genetic algorithms on the SOM models, optimising for the silhouette and SSI evaluation metrics, proved to be successful for extracting colours from images.

Image Segmentation Sub-Framework

The research question that the development of the image segmentation sub-framework addresses, "*Can the Ogma framework effectively segment images in an unsupervised manner?*", can be answered with a yes, as the image segmentation sub-framework was found to be able to effectively segment the images based on their contents.

Shape Extraction Sub-Framework

The Image Shape Ogma sub-framework was found to be successful at extracting circles from images. However, due to a lack of research into unsupervised methods for extracting shapes other than circles (squares, triangles, etc.), the extraction of these shapes was not incorporated into this sub-framework. This means that the research question which the development of this sub-framework addresses "*Can the Ogma framework effectively segment images in an unsupervised manner?*" can be answered affirmatively, but only with respect to circles.

Overall conclusions

The development of the Ence Ontology and the Ogma framework was mostly successful, however, the ineffectiveness of the image clustering sub-framework of the Ogma framework (and hence lack of labelling in the Ence ontology) represent significant issues with respect to the original aim of the development of the Ogma framework and Ence ontology.

9 Further Research

There are many ways in which the Ence Ontology and the Ogma framework and its components could be further developed, which will be discussed in this chapter.

Ence Ontology

An interesting way in which the Ence ontology could be further developed would be to have an adaptive schema (which would be generated by the Ogma framework) by using ontology learning methods. However, instead of using similar ontological learning methods as those applied in natural language processing, the methods followed would involve using feature descriptions within the image and matching these feature descriptions to other images to show how they are related [ontology learning from text]. The Ence ontology could also be updated to contain other properties of the images, such as the spatial relations, textures and shapes other than circles (i.e. e.g. pentagons, triangles) within images, and to contain HSV data. The search functionality of the ontology could be improved by bringing the Ogma framework's image clustering sub-framework to functionality, which would provide the ontology with accurately labelled images (as detailed in chapter 3) and thus allow for the intended search functionality of being able to request images that have been given the same label as that given to an uploaded image. Another way in which the search functionality could be further improved would be by clustering of segments of images, thus allowing for particular types of image segments to be searched on the ontology. Another way in which the Ence ontology search functionality could be extended would be by allowing for multiple images to be uploaded together and for results based on those images to be returned i.e. establishing a pattern across the images and returning results based on this pattern. Finally, the ontology could be further extended still to accommodate different types of data other than still images, such as gifs and video data, with the sequence of image frames being analysed.

Ogma Framework

Image Clustering Sub-Framework

The image clustering sub-framework was the only component of the Ogma framework which was deemed to not function effectively, due to the fact that it clustered nearly all images into the same cluster. A potential reason for this may have been the high-dimensionality of the data it was provided. A solution to this would be to perform dimensionality reduction on the input image dataset. Another issue may have been optimising for the silhouette evaluation when performing hyperparameter optimisation. One method by which the silhouette metric could have been made more appropriate for the evaluation of image data would be by using a convolved version of the metric, which would mean applying the convolution operation to images before providing them as input, or perhaps applying max-pooling or mean pooling to the images. Alternatively, a new evaluation metric could be invented to optimise the clustering sub-framework. Such a metric could potentially analyse clusters using sections of pixels as blocks to reduce the dimensionality, taking a 2x2 window and treating it as its own entity. A potential issue with this proposed method would be that rotated images of the same category would be put into different clusters, as this window would be subject to the rotation of the image if the dimensions of the image were odd. After being reconfigured to accurately cluster images for the purpose of label assignment, the clustering sub-framework could be extended to cluster the segments of images produced in chapter 6 The clustering sub-framework could also be extended to run the members of the genetic algorithms generations in parallel to reduce the amount of time taken to run the sub-framework.

Colour Extraction Sub-framework

The colour extraction sub-framework could be extended by further parallelising the genetic algorithms members such that each member runs as its own separate job, with the aim increasing the time in which generations are completed and thus achieving a faster execution time. The use of convolutional neural networks instead of the SOM algorithm in the framework could also be explored. Another way in which the colour extraction framework could be extended would be configuring it to be able to extract colours from non-still image data, such as gifs or videos.

Image Segmentation Ogma sub-framework

The image segmentation sub-framework presented in chapter 6 could be improved by developing a dataset that contains the ground truth segmentation data for the CIFAR-10 and Fashion-MNIST datasets. This would allow for further tuning of the CNN model by virtue of having a reference point to verify if images are being better segmented or not and for performance to be calculated in an automated fashion. Hyperparameter optimisation methods such as genetic algorithms could be applied in the tuning of the CNN segmentation model, optimising for mean intersection over union or some other evaluation metric. Other future work in relation to this sub-framework could include performing hyperparameter optimisation on the VOC2012 dataset, with the aim of maximising mean intersection over union.

Shape Extraction Ogma sub-framework

The shape extraction sub-framework could be further developed by using Markov random fields to attempt to extract shapes other than circles from within images (e.g. squares, triangles, polygons, etc.). Another method in which the shape extraction sub-framework could be further improved would be by utilising denoising techniques such as autoencoders to prevent background information being detected as a shape, so as to increase the likelihood of only having shapes that are significant to the image be detected and to allow for smaller circles and other shapes to be extracted.

Bibliography

[1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MA: MIT Press, 2016.

[2] T. Hastie, J. H. Friedman, and R. Tibshirani *The Elements of Statistical Learning* 2nd ed. NY: Springer, 2009.

[3] A. Krizhevsky, “Learning multiple layers of features from tiny images”, Technical report, University of Toronto, Toronto, 2009. [Online]. Available: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf [Accessed on: 1st Sep 2021].

[4] H. Xiao, K. Rasul and R. Vollgraf, “Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms”, Aug. 2017.

[5] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. “Imagenet: A large-scale hierarchical image database”. In Proc. CVPR, 2009.

[6] K. He, X. Zhang, S. Ren, and J. Sun. “Deep residual learning for image recognition”. In Proc CVPR, 2016.

[7] E. Hyvönen, A. Styrman, and S. Saarela. “Ontology-based image retrieval”. In Proceedings of XML Finland Conference, pp. 27-51, 2002.

[8] R. Szeliski, *Computer Vision: Algorithms and Applications*, NY: Springer, 2009.

[9] G. James, T. Hastie, D. Witten, R. Tibshirani *An Introduction to Statistical Learning: with Applications in R* NY: Springer, 2013.

[10] F. Chollet, *Deep Learning with Python.* New York: Manning, 2018.

[11] T. Kohonen, “Self-organized formation of topologically correct feature maps”. *Biol.* Cybern. 43, 59–69, 1982.

[12] P. J. Rousseeuw, “Silhouettes: a graphical aid to the interpretation and validation of cluster analysis,” J. Computational and Applied Mathematics, vol. 20, no. 1, pp. 53–65, 1987.

[13] V. Rajangam, N. Sangeetha, R. Karthik, and K. Mallikarjuna, “Performance Analysis of VGG19 Deep Learning Network Based Brain Image Fusion” In Handbook of Research on Deep Learning-Based Image Analysis Under Constrained and Unconstrained Environments, pp. 145-166, 2021.

[14] D. G. Lowe, “Object recognition from local scale-invariant features”. In Proceedings of the seventh IEEE international conference on computer vision vol. 20, pp. 1150-1157, Sep, 1999.

[15] G. E. Hinton, and T. J. Sejnowski, *Unsupervised learning: foundations of neural computation.* MA:MIT press, 1999.

[16] D. J. Kerbyson, and T. J. Atherton, “Circle detection using Hough transform filters”, 1995.

[17] S. J. K. Pedersen, “Circular hough transform” Vision, Graphics, and Interactive Systems, Aalborg University, Denmark, 2007.

[18] J. Xie, R. Girshick, and A. Farhadi, “Unsupervised deep embedding for clustering analysis”. In the International conference on machine learning, pp. 478-487 June, 2016.

[19] Scikit-Learn "scikit-learn Machine Learning in Python" 2021. [Online]. Available: https://scikit-learn.org/ [Accessed on: Sep. 1, 2021].

[20] Keras "Simple. Flexible. Powerful." 2021. [Online]. Available: https://keras.io/ [Accessed on: Sep. 1, 2021].

[21] Pytorch "FROM RESEARCH TO PRODUCTION" 2021. [Online]. Available: https://pytorch.org/ [Accessed on: Sep. 1, 2021].
[22] OpenCV "OpenCV is 20!" 2021. [Online]. Available: https://opencv.org/ [Accessed on: Sep. 1, 2021].
[23] rdflib "rdflib 6.0.0" 2021. [Online]. Available: https://rdflib.readthedocs.io/en/stable/ [Accessed on: Sep. 1, 2021].
[24] Free Software Foundation "GNU Parallel" 2021. [Online]. Available: https://www.gnu.org/software/parallel/ [Accessed on: Sep. 1, 2021].
[25] G. Vettigl "MiniSom" 2021. [Online]. Available: https://github.com/JustGlowing/minisom [Accessed on: Sep. 1, 2021].
[26] webcolors "webcolors 1.11.1" 2021. [Online]. Available: https://webcolors.readthedocs.io/en/1.11.1/ [Accessed on: Sep. 1, 2021].
[27] nmslib "Non-Metric Space Library (NMSLIB)" 2021. [Online]. Available: https://github.com/nmslib/nmslib [Accessed on: Sep. 1, 2021].
[28] scikit-image "Image processing in Python" 2021. [Online]. Available: https://scikit-image.org/ [Accessed on: Sep. 1, 2021].
[29] Owlready2 "Welcome to Owlready2's documentation!" 2021. [Online]. Available: https://owlready2.readthedocs.io/en/v0.32/ [Accessed on: Sep. 1, 2021].
[30] P. Siarry, *Metaheuristics.* Cham: Springer, 2013.
[31] Y. Lemoigne, and A. Caner,*Molecular Imaging: Computer Reconstruction and Practice* NY:Springer Science & Business Media, 2008.
[32] W. Kim, A. Kanezaki, and M. Tanaka, "Unsupervised learning of image segmentation based on differentiable feature clustering". IEEE Transactions on Image Processing, 29, pp.8055-8068, 2020.
[33] W3C "CSS Color Module Level 3" 2021. [Online]. Available: https://www.w3.org/TR/css-color-3/ [Accessed on: Sep. 1, 2021].
[34] M. Fang, G. Yue, and Q. Yu, "The study on an application of otsu method in canny operator." In Proceedings The 2009 International Symposium on Information Processing (ISIP 2009), China, p. 109., 2009.
[35] J. Hendler, F. Gandon, and D. Allemang, *Semantic Web for the Working Ontologist: Effective Modeling for Linked Data, RDFS, and OWL*, Morgan & Claypool, 2020.
[36] H. Schütze, C. D. Manning, and P. Raghavan, *Introduction to information retrieval*, Cambridge: Cambridge University Press, 2008.
[37] M. Douze, H. Jégou, , H. Sandhawalia, L. Amsaleg, and C. Schmid, "Evaluation of gist descriptors for web-scale image search." *In Proceedings of the ACM International Conference on Image and Video Retrieval*, pp. 1-8, July, 2009.
[38] S.A.K. Tareen, and Z. Saleem, "A comparative analysis of sift, surf, kaze, akaze, orb, and brisk." In 2018 International conference on computing, mathematics and engineering technologies (iCoMET) pp. 1-10, 2018, March.
[39] D. Bojanić, K. Bartol, T. Pribanić, T. Petković, Y.D. Donoso, and J.S. Mas, "On the comparison of classic and deep keypoint detector and descriptor methods." *In 2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA)* pp. 64-69, Sep 2019.
[40] K. Lenc, and A. Vedaldi, "Large scale evaluation of local image feature detectors on homography datasets", arXiv preprint arXiv:1807.07939 2018.
[41] N. Khan, B. McCane, and S. Mills, "Better than SIFT?", *Machine Vision and Applications*, 26(6), pp.819-836, 2015.

[42] D. Mistry, and A. Banerjee, “Comparison of feature detection and matching approaches: SIFT and SURF.” *GRD Journals-Global Research and Development Journal for Engineering*, 2(4), pp.7-13, 2017.
[43] M. Bennett, and K. Baclawski “The role of ontologies in linked data, big data and semantic web applications.” *Applied Ontology*, 12(3-4), pp.189-194, 2017.
[44] M. Aumüller, E. Bernhardsson, and A. Faithfull “ANN-benchmarks: A benchmarking tool for approximate nearest neighbor algorithms.” *In International Conference on Similarity Search and Applications* Springer, Cham, pp. 34-49. October 2017, October.
[45] L. Boytsov, and B. Naidan, “Engineering efficient and effective non-metric space library.” *In International Conference on Similarity Search and Applications* Springer, Berlin, Heidelberg, pp. 280-293, 2013, October.
[46] Y.A. Malkov, and D.A. Yashunin, “Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs.” *IEEE transactions on pattern analysis and machine intelligence*, vol. 42 no. 4, pp.824-836, 2018.
[47] Google Scholar "Learning multiple layers of features from tiny images" 2021. [Online]. Available: https://scholar.google.com/scholar?q=Learning+multiple+layers+of+features+from+tiny+images&hl=en&as_sdt=0&as_vis=1&oi=scholart [Accessed on: Sep. 1, 2021].
[48] H. Liu and L. Yu, “Toward integrating feature selection algorithms for classification and clustering.” *IEEE Transactions on knowledge and data engineering*, vol. 17 no. 4, pp.491-502, 2005.
[49] S. Alelyani, J. Tang, and H. Liu,"Feature selection for clustering: A review." *Data Clustering*, pp.29-60, 2018.
[50] S. Xiang, F. Nie, and C. Zhang, "Learning a Mahalanobis distance metric for data clustering and classification" *Pattern recognition*, vol. 41 no. 12, pp.3600-3612, 2008.
[51] J. MacQueen “Some methods for classification and analysis of multivariate observations” *In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability* Vol. 1, No. 14, pp. 281-297, June 1967.
[52] E. Xing, M. Jordan, S. J. Russell, and A. Ng, “Distance metric learning with application to clustering with side-information”. *Advances in neural information processing systems*, vol. *15*, pp.521-528, 2002.
[53] M. Steinbach, L. Ertöz, and V. Kumar, “The challenges of clustering high dimensional data.” *In New directions in statistical physics* Springer, Berlin, Heidelberg, pp. 273-309 2004.
[54] C. M. Bishop, *Pattern recognition and machine learning*, New York: Springer, 2006.
[55] De la Torre, Fernando and Kanade, Takeo. Discriminative cluster analysis. In ICML, 2006.
[56] U. Von Luxburg, “A tutorial on spectral clustering.” *Statistics and computing*, Vol. 17, no. 4, pp.395-416, 2007.
[57] M. Claesen. and B. De Moor. "Hyperparameter Search in Machine Learning" in MIC 2015: The XI Metaheuristics International Conference, Kasteelpark Arenberg 10, box 2446, 3001 Leuven, Belgium, 7-10 June 2015 pp. 14-1 – 14-5.
[58] L. M. R. Rere, M. I. Fanany, and A. M. Arymurthy, "Metaheuristic Algorithms for Convolution Neural Network" *Computational Intelligence and Neuroscience,* vol. 2016, Article ID 1537325, pp. 1-13, 2016.
[59] C. C. Aggarwal, A. Hinneburg, and D.A. Keim, “On the surprising behavior of distance metrics in high dimensional space.” *In International conference on database theory* Springer, Berlin, Heidelberg, pp. 420-434, January 2001.

[60] J. R. Koza, "Genetic Programming" *Encyclopedia of Computer Science and Technology,* pp 1-26 Aug. 1997.
[61] G. Harik, F. Lobo, and D. Goldberg, "The compact genetic algorithm," IEEE Transactions on Evolutionary Computation, vol. 3, no. 4, pp. 287– 297, November 1999.
[62] L. Wang, S. Xie, T. Li, R. Fonseca, and Y. Tian, “Sample-efficient neural architecture search by learning actions for monte carlo tree search.” *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 2021.
[63] P. Foret, A. Kleiner, H. Mobahi, and B. Neyshabur, “Sharpness-aware minimization for efficiently improving generalization.” arXiv preprint arXiv:2010.01412, 2020.
[64] X. Wang, D. Kihara, J. Luo, and G.J. Qi, “EnAET: A self-trained framework for semi-supervised and supervised learning with ensemble transformations.” *IEEE Transactions on Image Processing*, vol. 30, pp.1639-1647, 2020.
[65] A. Nøkland and L.H. Eidnes “Training neural networks with local error signals.” *In International Conference on Machine Learning* pp. 4839-4850, May, 2019.
[66] Y. Sun, L. Zhang, and H. Schaeffer, “Neupde: Neural network based ordinary and partial differential equations for modeling time-dependent data.” *In Mathematical and Scientific Machine Learning* pp. 352-372, August, 2020.
[67] O.C. Granmo, S. Glimsdal, L. Jiao, M. Goodwin, C.W. Omlin, and G.T. Berge, “The convolutional Tsetlin machine.” arXiv preprint arXiv:1905.09688, 2019.
[68] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition.” *In Proceedings of the IEEE conference on computer vision and pattern recognition* pp. 770-778, 2016.
[69] Mahon, L. and Lukasiewicz, T., Selective Pseudo-label Clustering. arXiv preprint arXiv:2107.10692, 2021.
[70] M. Willetts, S. Roberts, and C. Holmes, “Semi-Unsupervised Learning: Clustering and Classifying using Ultra-Sparse Labels.” *In 2020 IEEE International Conference on Big Data (Big Data)* pp. 5286-5295, December, 2020.
[71] N. Mrabah, N.M. Khan, R. Ksantini, and Z. Lachiri, “Deep clustering with a dynamic autoencoder: From reconstruction towards centroids construction.” *Neural Networks*, vol. 130, pp.206-228, 2020.
[72] S. Luke, *Essentials of Metaheuristics* 2nd ed. California :Lulu, 2009.
[73] archive.org "Partitioning Around Medoids (PAM)" 2021. [Online]. Available: https://web.archive.org/web/20111002220803/http://www.unesco.org:80/webworld/idams/advguide/Chapt7_1_1.htm [Accessed on: Sep. 1, 2021].
[74] Venugopal, K.R. and Patnaik, L.M. eds., 2012. Wireless Networks and Computational Intelligence: 6th International Conference on Information Processing, ICIP 2012, Bangalore, India, August 10-12, 2012. Proceedings (Vol. 292). Springer.
[75] G. Sharma, and R. Bala *Digital color imaging handbook*. Ardenne: CRC press, 2017.
[76] G. Ramella, “Evaluation of quality measures for color quantization.” *Multimedia Tools and Applications*, pp.1-35, 2021.
[77] P. Heckbert, “Color image quantization for frame buffer display.” *ACM Siggraph Computer Graphics*, vol. 16 no. 3, pp.297-307, 1982.
[78] J.C. Bezdek, R. Ehrlich, and W. Full, *FCM: The fuzzy c-means clustering algorithm.* Computers & geosciences, vol. 10 no. 2-3, pp.191-203, 1984.
[79] M.E. Celebi, “Improving the performance of k-means for color quantization.” *Image and Vision Computing*, vol. 29 no. 4, pp.260-271, 2011.
[80] blank purposefully
[81] D.E Rumelhart, J.L. McClelland, and PDP Research Group, “Parallel distributed processing” vol. 1, pp. 354-362, Massachusetts, 1988.

[82] A. Koschan, and M. Abidi, *Digital color image processing*. John Wiley & Sons, 2008.
[83] T. Kohonen, “Self-organizing maps: ophmization approaches.” *In Artificial neural networks* pp. 981-990 North-Holland, 1991.
[84] N. Thomos, N.V. Boulgouris, and M.G. Strintzis, “Optimized transmission of JPEG2000 streams over wireless channels.” *IEEE Transactions on image processing*, vol. 15 no. 1, pp.54-67, 2005.
[85] Z. Wang, and A.C. Bovik, “Mean squared error: Love it or leave it? A new look at signal fidelity measures.” *IEEE signal processing magazine*, vol. 26 no. 1, pp.98-117, 2009.
[86] A. Kanezaki, “Unsupervised image segmentation by backpropagation.” *In 2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)* pp. 1543-1547, April, 2018.
[87] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, “Generalized intersection over union: A metric and a loss for bounding box regression.” *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* pp. 658-666, 2019.
[88] C.R. Brice, and C.L. Fennema, “Scene analysis using regions”. *Artificial intelligence*, vol. 1 no. 3-4, pp.205-226, 1970.
[89] M. W. Pettet “Shape and contour detection” *Vision Research*, vol. 39 no. 3, pp.551-557, 1999.
[90] X. Yu, Y. Rao, Z. Wang, Z. Liu, J. Lu, and J. Zhou, “PoinTr: Diverse Point Cloud Completion with Geometry-Aware Transformers.” arXiv preprint arXiv:2108.08839, 2021.
[91] H. Xie, H. Yao, S. Zhou, J. Mao, S. Zhang, and W. Sun, “Grnet: Gridding residual network for dense point cloud completion.” *In European Conference on Computer Vision* pp. 365-381, Springer, Cham, August, 2020.
[92] W. Yuan, T. Khot, D. Held, C. Mertz, and M. Hebert, “Pcn: Point completion network.” *In 2018 International Conference on 3D Vision (3DV)* pp. 728-737, September 2018.
[93] A.O. Djekoune, K. Messaoudi, and M. Belhocine, “A New Modified Hough Transform Method for Circle Detection” *In IJCCI* pp. 5-12, September, 2013.

A1 Appendix

A1.1 Appendix A

The datasets that are used in the creation of this dissertation are:

- The Fashion-MNIST dataset, which was developed by Zalando Research and is licensed under the MIT license. The dataset contains a predefined train and test split and a total of 70,000 images which belong to 10 classes. This predefined train and test data split was used for the purposes of this dissertation. The dataset training set has a total of 60,000 images and 10,000 belong to the test set in the Fashion-MNIST dataset.
- The CIFAR-10 which was created by Alexander Krizhevsky is available under the Creative Commons Attribution 4.0 License, the dataset contains a total of 60,000 images which belong to 10 classes. This predefined train and test data split was used for the purposes of this dissertation. The dataset training set has a total of 50,000 images and 10,000 belong to the test set in the Fashion CIFAR-10 dataset.

A1.2 Appendix B

A video demonstration of the Ence knowledge graph populated with Fashion-MNIST data extracted using the Ogma framework can be found at:
This video demonstration showcases the knowledge graph being searched using another image, the results being returned by that query.
Additionally this video also shows the ontology being searched with colour data.

A1.3 Appendix C

The supplementary materials for this submission are attached in a zip folder called: "20300057-joshua-cassidy-CS7CS5-dissertation-supplementry-materials.zip". For instructions on how to run the project see the readme.md file in the root directory.

A1.4 Appendix D

The test cases that were developed to ensure this dissertation's Genetic algorithms functionalities in the Ogma framework operated correctly and as expected, a sample of the important test cases were selected to highlight the key pieces of functionality that were tested. The instructions to run these test cases can be found in readme.md file located in the dissertation code base in appendix C

Test Colour Extraction Sub-framework Mutation Operator

Test Case ID

OTC01

Test Type

Automated unit test

Test Case Description

The functionality which this test case tests is the mutation operator in the colour extraction Ogma sub-framework by way of automated unit tests (using the unittest python testing library). The mutation operator is responsible for mutating a SOM population individuals genes via mutation.

Pre-Conditions

a. Mock the mutation rate and the crossover rate
b. Set a random seed

Execution Steps

1. Create a blank SOM population individual object
2. Assign the clusters, topology, neighbourhood and sigma genes to the newly created SOM population individual
3. Perform the mutation operation on the SOM population individual
4. Assert the mutated SOM population individual genes are equal to the expected SOM population individuals mutated genes

Post-Conditions

None

Test Status

Passed

Test Colour Extraction Sub-framework Crossover Operator

Test Case ID

OTC02

Test Type

Automated unit test

Test Case Description

The functionality which this test case tests is the crossover operator in the colour extraction Ogma sub-framework by way of automated unit tests (using the unittest python testing library). The crossover operator is responsible for creating a child by mixing two parents genes together.

Pre-Conditions

a. Mock the mutation rate and the crossover rate
b. Set a random seed

Execution Steps

1. Create 2 blank parent SOM population individual objects
2. Assign the clusters, topology, neighbourhood and sigma genes to the newly created SOM population individual parent objects
3. Perform the crossover operation on the SOM population individual parent objects and assign the output to be a new child object
4. Assert the newly created SOM population individual child object's genes are equal to the expected genes that would be created from the two parent objects

Post Conditions

None

Test Status

Passed

Test Colour Extraction Sub-framework Selection Operator

Test Case ID

OTC03

Test Type

Automated unit test

Test Case Description

The functionality which this test case tests is the selection operator in the colour extraction Ogma sub-framework by way of automated unit tests (using the unittest python testing library).

The selection operator is responsible for selecting a population member to be used in the crossover operator in the creation of the next generation

Pre-Condition

a. Mock the mutation rate and the crossover rate
b. Set a random seed

Execution Steps

1. Create a population of blank SOM population individual objects
2. Assign the a fitness value to each of the newly created SOM individuals in the created population
3. Perform the selection operator to select a SOM individual from the population
4. Assert the fitness value of the selected SOM individual from the population is equal to the expected fitness value

Post-Conditions

None

Test Status

Passed