THE ASSOCIATION BETWEEN SENTIMENT AND COLOUR USAGE OF IMAGES FROM INSTAGRAM, INVESTIGATED THROUGH INNOVATIVE DATA VISULAISATION TOOLS

A dissertation submitted to The University of Manchester for the degree of Master of Science MSc Business Analytics: Operational Research and Risk Analysis in the Faculty of Humanities

2020

10480961

Alliance Manchester Business School

List of Contents

Word Count (main body): 13,019

List of Figures

List of Tables

Abstract

An increasing number of photo-centered social media platforms are taking charge of people's life and providing diversified visual contents revealing the nature of social relationship. Sentiment analysis techniques have been developed to understand the emotional content of text, whereas a limited number of studies have probed into the sentiment delivered through visual elements. Here, we draw images from the social media site, Instagram, and use the clustering technique to classify images into groups according to their hashtag properties. We propose a novel data visualisation tool to display the image data points in a three-dimensional virtual reality space for colour observation, aiming to figure out the association between colour patterns and semantic properties of each group more intuitively and to evaluate the effectiveness of the visualisation method. The HSV colour analysis results affirm the highest saturation and value of images attached with only positive hashtags and give insights into the colour patterns in images of other semantic features. The data visualisation method is in favour of more detailed colour pattern discovery and can be applied to support other research in fields like this.

Intellectual Property Statement

I. The author of this dissertation (including any appendices and/or schedules to this dissertation) owns certain copyright or related rights in it (the "Copyright") and s/he has given The University of Manchester certain rights to use such Copyright, including for administrative purposes.

II. Copies of this dissertation, either in full or in extracts and whether in hard or electronic copy, may be made only in accordance with the Copyright, Designs and Patents Act 1988 (as amended) and regulations issued under it or, where appropriate, in accordance with licensing agreements which the University has entered into. This page must form part of any such copies made.

III. The ownership of certain Copyright, patents, designs, trademarks and other intellectual property (the "Intellectual Property") and any reproductions of copyright works in the dissertation, for example graphs and tables ("Reproductions"), which may be described in this dissertation, may not be owned by the author and may be owned by third parties. Such Intellectual Property and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property and/or Reproductions.

IV. Further information on the conditions under which disclosure, publication and commercialisation of this dissertation, the Copyright and any Intellectual Property and/or Reproductions described in it may take place is available in the University IP Policy, in any relevant Dissertation restriction declarations deposited in the University Library, and The University Library's regulations.

Acknowledgement

On the very outset of this dissertation, I would like to extend my heartiest gratitude towards all the personage who have helped me in this endeavor.

Thanks to my supervisor, **Professor Nikolay Mehandjiev**, for inspiring my interest in the development of innovative technologies and providing guidance through each stage of the process.

Thanks to **Dr Qudamah Quboa** and **Dr Zixu Liu** for assisting me with the implementation of data visualisation in the Unity.

I am grateful to the Alliance Manchester Business School for providing the state of the art of virtual reality technology in the Data Visualisation Observatory for me to gain impressive experience during the completion of this project.

Thanking you

Yi-Hsuan, Chen

Section 1: Introduction

Sharing posts has become one of the most popular activities in primary social network services for people to express their opinions and emotions to link up a friendship and social support [1]. According to existing surveys, multimedia contents become more prevalent in recent days, which makes visual communication an essential part of the lives of billions of people and forms the basis of many prominent photo-centred social media systems such as Instagram, Pinterest and Twitter [2]. In these platforms, posts mainly consist of images and short tags or even with no text at all, and thus lots of self-representation and self-expression are conveyed through visual contents alone.

It is crucial to understand the communicative sentiment in assessing its affective qualities, and thus there has been emerging interest in data mining for automatic sentiment detection [3]. However, most of the computational analysis concentrates on sentiment delivered by text, while limited efforts have been made to analyse sentiment from visual content such as images and videos, which is becoming a pervasive media type on the web [4].

In the society where visual experience hugely associated with human decision and behaviour, how people communicate feelings through colour and image structures is worthwhile to be explored. In this project, We mainly focus on the low-level feature, colour [5], of images and take the attached hashtags as standardized expressions that may capture sentiment, aiming to figure out the relationship between image colour usage and the emotions delivered by hashtags on the social media platform. Given that psychovisual expression of sentiment can vary with

social communication contexts, We restrict attention to images from the social media platform Instagram, which is also typically used for self-representation and relationships [6].

On the other hand, the human brain processes images faster than text, and most of the information transmitted to the brain is visual. An adequate data visualization tool can help us enhance data processing and organizational effectiveness, letting us probe into a problem without the frame of a conjecture [7]. Hence, a three-dimensional visualisation technique was introduced for displaying image data and gaining more comprehensive insight into the research problems.

In this research, We performed a series of language processing steps to format the hashtags into unique word labels [8] and use the Hierarchical clustering to classify images by their labels. During the clustering, a dimensional reduction method was applied to the sample data for colour observation in 3D space. Eventually, a colour analysis was conduct on images of each group to find specific colour patterns and make evaluations for the visualisation tool used.

Section 2: Background and Literature Review

The topic of this project is related to two different research areas, colour emotions on an image-based social media platform and the evaluation of three-dimensional data visualisation tools for research exploration. The relation between image colour and emotions and the tools used for analysis had been discussed in several previous studies.

Solli and Lenz (2008) experimented with five-thousand images and showed that colour emotion properties are well suited for semantic image classification [9]. They introduced a colour emotion concept derived from psychophysical experiments which described colour through three emotion scales: activity, weight and heat. In their study, Solli and Lenz extracted colours from image pixels in the CIELAB colour system format and converted them into semantic-degree expression, which became the image's colour emotion features for training the image retrieval system. However, during the demonstration, they scattered images in three two-dimensional plots to show images' relationship with colour emotion properties, which was not intuitive enough for the observation of colour distribution.

Borth et al. (2013) addressed the challenge of sentiment analysis from visual content [5]. They created more than 3,000 adjective-noun pairs (ANP) according to the sentiment words defined in Plutchik's theory[1] and served these as the semantic concepts of images collected from Flickr. They utilised two popular

[1] Plutchik's theory is a psychoevolutionary classification approach for general emotional responses. Plutchik suggested that there were eight primary emotions - anger, fear, sadness, disgust, surprise, anticipation, trust, and joy.

linguistics-based models, SentiStrength and SentiWordNet, to quantify the sentiment values of ANPs and used them to train the image detectors. When creating ANPs, they also identified and addressed the issue of the pairs including an adjective and a noun with opposite sentiment value, such as “abused” and “child”, which tend to reflect stronger negative sentiment. In this research, they considered mid-level features of images, such as the semantic representation, more than low-level features, such as colour, while we aim to maintain the exploration of both aspects to a balanced degree.

The two studies mentioned above probed into how colour or images linked to different emotions based on existing theoretical classification methods and assumptions, while did not discuss how people practically used these visual contents when expressing their feelings. We combined the concept of the semantic representation in the study of Borth, et al. and the techniques used by Solli and Lenz to extract colour from image pixels, aiming to acquire insight into social media users’ colour usage habits in the current generation.

In Amencherla and Varshney’s work (2017), they pointed out the importance of visual communication in today’s highly social-media-based life and performed a similar study as ours [10]. They collected three groups of images from Instagram that were tagged with *happy*, *awesome* and *sad*, respectively, and conducted a hypothesis test on those images with colour psychological theories drawn from related literature. The results confirmed that there was a significant colour difference between images with tag *sad* and those with *happy* or *awesome*; they also found some associations between colour and sentiment that did not correspond to the earlier colour psychological findings.

This research took hashtags as the properties of images. However, it barely

concerned that posts with tag *happy* could also be tagged with *awesome* or other semantic hashtags that expressed sentiment as well. Those other hashtags could have an even stronger association with the image colour presentation, in which case the observed colour reflect mainly the sentiment of other tags rather than *happy*, *awesome*, and *sad* themselves. On the other hand, some Instagram users tend to mix positive and negative hashtags in a post, for example, *love* with *sad*, or even *awesome* could be with *sad*. In this case, users often wanted to express negative feelings more than the positive ones; the similar concerns had been discussed in the study of Borth et al [5].

Amencherla and Varshney's study reflected an issue that the traditional colour psychophysical theories might not entirely explain the colour used on the current social media platform. Most of the existing prominent colour research were either developed years ago or were conducted under strictly controlled situations [11], and these could be the reasons for the lack of interpretability of colour theories on today's intricate social media ecosystems. We try to find out image colour patterns through clustering their related semantic words given by social media users instead of directly classifying images by colour properties and hope to look into how people today use colour in daily life to deliver emotions and what kind of emotions different colour means to them.

Colour is a highly intuitive concept that directly influences our thoughts through our eyes; hence introducing a proper visualisation technique during the investigation and demonstration stage is vital for colour-related exploration. Since multi-dimensional feature vectors were used in our project, techniques such as 3D visualisation played a significant role for us to navigate through the high-dimensional space and reveal the relationship among data points.

To our knowledge, scarce earlier attempts are made to use three-dimensional visualisation tools for image data display. However, a team of Google Brain developed a web application called Embedding Projector, which is an interactive tool that shows the embedding[2] results of input data [12]. The tool can perform dimensionality reduction by either Principal Component Analysis (PCA) or t-SNE technique and visualise high-dimensional data in 2D or 3D space through Python TensorBoard platform. We realise the image demonstration in 3D space by replacing the data points by images in TensorBoard and follow the same idea to reproduce the visualisation in a 3D game engine software, Unity, for further evaluation.

[2] An embedding is a map from input data to points in Euclidean space.

Section 3: Research Methodology and Methods

This research aimed to gain insight into the association between image colour usage and human emotions expressed by hashtags on the social media platform, Instagram, and to explore how a proper visualisation tool can support research in fields like this. During the research, a large amount of data was gathered from Instagram's user interfaces at the beginning of the project, and the whole data processing procedure was done through the online programming tool, Python Jupyter Notebook.

We drew the general data analysis methodology from the relevant literature [13] and developed our project according to the process of the "Unsupervised" path illustrated in **Figure 1**. However, to also emphasise on the role visualisation tools played in our research, the methodology structure had been somewhat adjusted into the following order:

- *Data Collection and Protection*
- *Data Preprocessing*
- *Data Visualisation for Exploration*
- *Determining and Performing Data Mining Tasks* and, finally
- *Interpreting the Results and Deriving Insight*

The first four stages above were covered in this section, and the last one would be later discussed in Section 4 and 5.

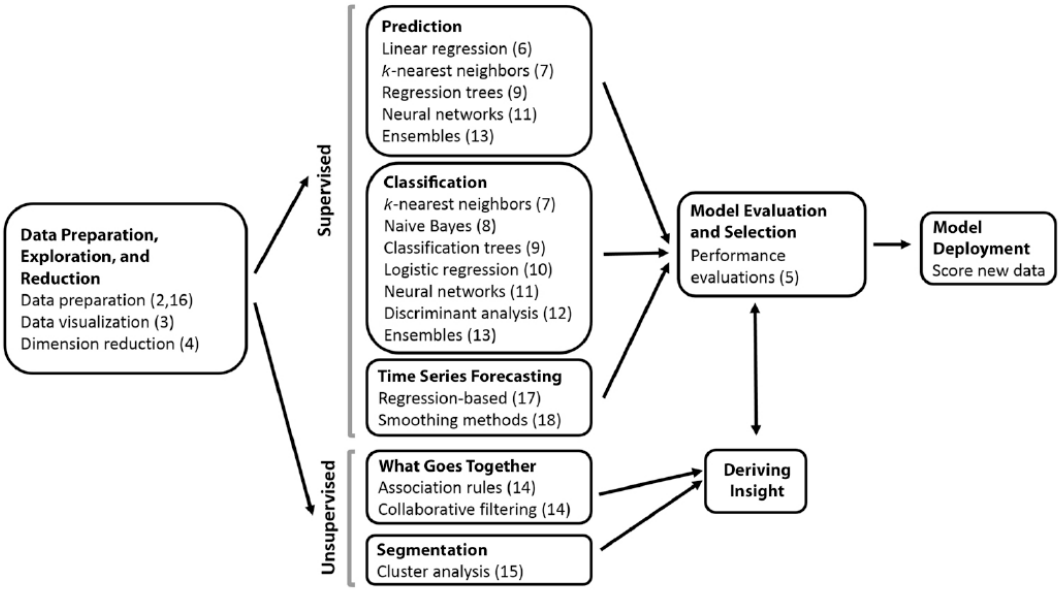


Figure 1: Data mining from a process perspective

3.1. Data Collection and Protection

3.1.1. Data Collection

3.1.1.1. An Overview

The two essential elements demanded from the Instagram platform were images and the attached hashtags in a large quantity of posts.

In an extensive research by Gao et al. [14], it was concluded that the impact of cultural background on colour emotions was very limited; therefore we excluded the considerations of variation caused by location difference when collecting data and randomly gathered posts based on several chosen hashtags. For example, if the hashtag “happy” were selected, we would gather lots of posts linked with “happy” and extract the required contents from them to enrich our raw database.

In order to increase the sentiment diversity, sixteen initial hashtags were chosen as the filter conditions during the collection stage. When choosing initial hashtags, we drew support from the opinion mining tool, SentiStrength [15], and selected eight hashtags with positive sentiment values, and the other eight ones with negative sentiment values.

The way we decided the initial hashtags was to look up top-used Instagram hashtags from the website: https://top-hashtags.com/instagram/ and, at first, randomly chose eight positive ones linked with more than a hundred million posts. The chosen hashtags are *love*, *beautiful*, *smile*, *amazing*, *inspiration*, *delicious*, *happiness* and *relax*. Since people less frequently used negative hashtags according to the website, we lowered the threshold and chose eight negative ones linked with more than just one million posts; these hashtags were *sad*, *cry*, *depressed*, *upset*, *hate*, *tears*, *stressed*, *heartbroken*.

We constructed a loop and ran sixteen iterations to gather posts containing each of the sixteen initial hashtags through the Instagram public API[3]. The API could call a real-time JSON[4] page involving anonymous plain-text information about the platform's public feeds which included the owner ID, timestamps, like counts, any types of text contents, such as captions and hashtags, the image URLs of each post and the like. We then extracted the owner IDs, images and hashtags into a prepared data frame in the Jupyter Notebook. However, the acquisition of some images and hashtags was not successful due to the internet errors and other internal permission issues; in this case, the unobtainable data was marked with "Missing" in the *image_URL* or *hashtag* column of the data frame.

[3] Application Programming Interface
[4] JavaScript Object Notation

During the data collection stage, 6,236 sets of raw data were collected through positive initial hashtags, while 7,425 sets of raw data were collected through negative initial hashtags. These two groups of data were marked with sequential numbers and alphabets, respectively, in the *mark* column of the data frame, to be monitored the data changes in the next stage.

The final data frame at this stage consisted of four columns, which were *owner_ID*, *image_URL*, *hashtag* and *mark*. This data frame was named as *RawData* for further processing and exported as a CSV file as a record.

3.1.2. Data Protection and Storage

All the data used in our project was collected from the public accounts of Instagram, and the owners of those accounts had consented to share the contents on their feeds to the general public in accordance with Instagram's conditions and terms. The owner ID consisted of randomly generated numbers that Instagram used to identify users by the company's internal algorithms, while general people were not able to acquire any further information from it. During the period of obtaining data, the researchers were not accessible to the account owners' personal details, such as their names, sex, birth and locations; the collected raw data was appropriately stored in researchers' personal drive and the University's encrypted cloud storage to be processed. For the image data, precisely, no image was reproduced in its original form in this dissertation, and all of them would be heavily pixelated to ensure anonymity on display.

Once the research period is over, the raw data will be destroyed, and the research results will also be uploaded to the secured research data storage of the Data

Visualisation Observatory of the University of Manchester, to be used for demonstrating the facilities of the lab.

3.2. Data Preprocessing

3.2.1. Data Cleansing

3.2.1.1. Dealing with Missing Values

In the *RawData* data frame, it was found that there were several sets of data marked with "Missing" in *image_URL* and *hashtag* columns. Since both images and hashtags were indispensable variables to our project, we avoided missing values by directly removing rows containing the "Missing" mark in either of the columns.

After dropping the data with missing values, 12,802 sets of data were remained, among which 5,779 sets of data were collected from positive initial hashtags, and 7,023 of them were collected from negative initial hashtags, according to the *mark* column.

3.2.1.2. Dropping Duplicated Data

Practically, a post in Instagram feeds could contain the hashtags, for example, *love* and *happiness* at the same time. If we used these two hashtags to collect one-hundred posts, respectively, there might be several identical posts gathered from the same owners. The way to drop these duplicates was to ensure that there were no duplicated values in the *owner_ID* column of the *RawData* data frame; we kept rows with the first-appeared values in the *owner_ID* column and dropped the rows with duplicated ones. While by using this method, different posts with the same

owners might also be ruled out, it could avoid gathering perspectives from the same Instagram users and help maintain data variety.

On the other hand, it was found that different account owners might include the same image sources in their posts; in this case, we followed the same method and kept only rows with the first-appeared values in the *image_URL* column of the *RawData* data frame to reduce the repeatability of visual contents in the data visualisation stage.

After dropping the duplicates, 7,785 sets of data were remained, among which 3,760 sets of data were collected from positive initial hashtags, and 4,025 of them were collected from negative initial hashtags, according to the *mark* column.

In the final step of this stage, we removed the *owner_ID* column and assigned a new *ID* column containing the sequential reset numbering to the whole data frame. The values in *ID* column ranged from 0 to 7784 and were used for identification during the rest of the data processing stages. The data frame was now containing four columns, which were *ID*, *image_URL*, *hashtag*, *mark*, and was given a new name, *Allinfo*, for further processing.

3.2.1.3. Image Labelling and Filtering

In order to inspect the quality of raw image data and process images in the Jupyter Notebook, the 7,785 images were downloaded through the image URLs and named after their corresponding data IDs according to the *Allinfo* data frame. All downloaded images were formatted into the regular size with 217 pixels in width and 217 pixels in length and were stored into an encrypted folder for use.

The aim of this project was to explore the relationship between colour used and

sentiment feelings people intended to express; hence, images containing too many text contents were not ideal for the analysis. Furthermore, Instagram users often used cartoon or animation screencaps as the images shown in their posts; these images were not the photos taken by users themselves and might not authentically reveal how colours related to their daily life, which was also undesirable for our project. Therefore, we set criteria to ensure that (1) the text contents in the image and the background-area for bringing out the text should not account for more than one-fifth of the image layout, (2) the text contents should not be located in the middle third area of the image (either in vertical or horizontal way), and (3) the contents of images should include real-world creatures or scenes.

Instagram users displayed text contents in images in a wide variety of ways, such as using unusual fonts or adding visual effects. To raise the correctness rate of distinguishing images against the three criteria above, we decided to manually screen out unqualified images from the 7,785 sets of image data by eyeball test.

During this process, 2,388 images were filtered out, and the number of the remaining image data was 5,397, among which 2,893 images were collected from positive initial hashtags, and 2,504 images were from negative initial hashtags.

To synchronize datasets stored in the *Allinfo* data frame, we recorded the file names, which was also the data IDs, of the dropped images, and removed the rows with the same IDs in *Allinfo* data frame. After the adjustment, the number of the datasets in *Allinfo* became 5,397 at the end of this stage.

3.2.2. Hashtag Preprocessing

3.2.2.1. An Overview

Hashtags are creative labels used in micro-blogs to characterize the topic of a message or discussion [16]. These labels could be made up of freely-combined characters, such as letters, digits or symbols, without length limit, and were all preceded by a hash sign (#). We performed a series of language processing methods to uniform these unstructured labels in the *hashtag* column of the *Allinfo* data frame and added a new column, *preprocessed_htag*, to store the processed results. We also identified the sentiment value of each word to link images with emotional hashtags. Meanwhile, to monitor the changes, we calculated the number of the unique hashtags existing in the whole dataset and recorded them into a list at the end of each preprocessing stage.

3.2.2.2. Converting to Lowercase & Removing Digits and Punctuation

For Python programming language, the string *Love* and *love* were different objects while they actually had the same meaning; thus, we turned all letters in the hashtags into lowercase to prevent error recognition. Instagram users sometimes put punctuation in their hashtags for aesthetic reasons or used digits to express particular meanings; since we considered only the emotions pure words conveyed, all the punctuation and digits included in the hashtags were removed. The results of the procedures above were stored into the *preprocessed_htag* column of the *Allinfo* data frame. At the end of this stage, the number of unique hashtags was 32,213.

3.2.2.3. Word Segmentation

Some hashtags consisted of a sentence without space, such as *loveandpeace* from "love and peace" and *bestoftheday* from "best of the day". These hashtags might contain words with sentiment value, but the opinion mining tools could not easily detect them. Hence, we split this kind of hashtags by using the Python module, WordSegment, which was an Apache2 licensed [5] module for English word segmentation and was written based on a database derived from the Google Web Trillion Word Corpus. Within this module, every base word and phrase is lowercased with punctuation removed.

After conducting word segmentation on the *preprocessed_htag* column of the *Allinfo* data frame, lots of the split-out words happened to be identical to the existing hashtag words. For example, *loveandpeace* was split into "love", "and" and "peace", while each of these three hashtags might already exist in the original dataset; in this case, they were not counted again into unique hashtags. By this reason, the number of unique hashtags reduced to 17,420 at the end of this stage.

3.2.2.4. Removing Words of Unwanted Part of Speech

Due to the word segmentation, several hashtags like "of", "and", "the" were created. These words were somewhat meaningless when appearing alone, and they mostly belonged to the part of speech such as preposition or conjunction. To include only meaningful hashtags that could deliver human thoughts, the Natural Language Toolkit (NLTK), a suite of programs written in Python for symbolic and statistical natural language processing, was used to filter out words of unwanted

[5] Apache License is a permissive free software license written by the Apache Software Foundation (ASF) and allows users to use the software for any purpose without concern for royalties.

parts of speech. The *nltk.pos_tag* function in NLTK labelled the words (hashtags) with their most-frequently-used part of speech and returned the results in an array. We then constructed a loop, running through the *preprocessed_htag* column, and dropped hashtags not belonging to verbs, adjectives or nouns. The number of unique hashtags slightly decreased to 17,313 at the end of this stage.

3.2.2.5. Word Lemmatization

During the formation of hashtags, Instagram users tend to modify the words to express different grammatical categories, such as tense, case, voice and the like, while words of different grammatical types often refer to the same emotional meanings. To uniform these words into the same type, we used the lemmatization technique to reduce the inflected words, properly ensuring that the root word, which was called lemma, belonged to the language. For example, smiles, smiling, smiled are all forms of the word smile, and therefore they would be all converted into their common lemma, smile.

In the Jupyter Notebook, the package Natural Language Toolkit (NLTK) was used to conduct this process. NLTK provided the *WordNetLemmatizer()* function that used the WordNet database[6] to look up lemmas of words; this function had a parameter *pos*, which referred to the parts-of-speech and controlled the grammatical categories the input words were to be converted into. For example, if we assigned "v" (verb) to *pos*, the function would transfer "amazing" to "amaze", "depressed" to "depress", while remaining "tears" as "tears" since there was no verb-form lemma for "tears". Similarly, if we assigned "a" (adjective) to *pos*, the

[6] The WordNet database is a lexical database of semantic relations between words developed by Princeton University.

function would restore the comparative and superlative forms of an adjective, transferring “happier” and “happiest” to “happy”; if we assigned “n” (noun), “tears” became “tear”.

The lemmatization order in this process was first to convert all adjectives, and then nouns, and finally verbs. It was worth notice that some of the adjectives with suffixes “ed” or “ing” were transferred into verb form during the final conversion. The number of unique hashtags at the end of this stage reduced to 15,395.

3.2.2.6. Filtering Out Non-English Words

In the previous hashtag preprocessing steps, all changes were applied to the *Allinfo* data frame, while in this step, we performed changes only on the unique hashtag list. The current unique hashtag list contained 15,395 different words, and some of them were not vocabularies defined in standard English dictionaries. The possible reasons for the appearance of undefined words were spelling errors, words from non-English languages or that words were arbitrarily created. Since a word like “loveandpeace” could be recognized as any of the situations above, performing word segmentation before processing undefined English words allowed us to keep more meaningful terms to analyze.

The word corpora in the NLTK package was used to identify the “non-English words”. NLTK contained several English corpora such as *gutenburg*, *brown*, *reuters* and *inaugural*, and all of them were language resources consisting of an extensive collection of structured texts from various electronic platforms. The in-built *words.words()* function in NLTK package helped us check if a word was from any of the corpora stated above; we then dropped the non-English words determined by this function from the unique hashtag list. At the end of this stage,

the number of unique hashtags reduced more than half to 6,493.

3.2.2.7. Assigning Sentiment Values

After dropping the non-English words, we assigned sentiment values to each of the remaining 6,493 words in the unique hashtag list. In this stage, a new data frame, *HtagSenti*, was created to store the processing results. The tool used to define the sentiment values was SentiStrength, a free sentiment analysis program developed by Dr Michael Thelwall, evaluated and applied in several peer-reviewed academic articles and research projects[7].

SentiStrength used a lexical approach to classify social web texts in dual positive – negative scales because psychological studies reported that humans could experience positive and negative emotions simultaneously and to some extent independently [15]. We downloaded the SentiStrength java file authorized by Dr Thelwall and linked it to Python Jupyter Notebook to import *PySentiStr()* function for use. The function allowed us to choose four presentation ways of the sentiment values, which were *dual*, *binary*, *trinary* and *single* scales. *Dual* scale gave both the positive value, from 1 to 5, and negative value, from -1 to -5, to the estimated content, and was the default setting of the function. *Binary* returned the value 1 if the estimated word was positive and -1 if its negative; *trinary*, besides positive and negative values, also provided a neutral score of the word; *single* scale added up the positive and negative values together and returned a single estimate value, from -4 to +4, of a word.

The *single* scale was used in our project to acquire the sentiment values since it

[7] The list of articles and projects including SentiStrength in the research were recorded in the following website: http://sentistrength.wlv.ac.uk/

could provide more overall and intuitive estimates of the words for classification. In the *HtagSenti* data frame, two columns were created; the *uni_htag* column stored the unique hashtags and the *senti_value* column store the hashtags' corresponding sentiment scores. Among the *HtagSenti* data frame, it was found that many hashtags were with no sentiment, which means having a sentiment value equaling zero, such as the words "female", "active", "instant" and so forth. Seeing that our goal was to find the association between image colour and emotion elements, the rows of *HtagSenti* with a zero value in the *senti_value* column were kept out. At the end of this stage, the final quantity of the unique hashtags drastically reduced to 835, which was also the number of rows in *HtagSenti* data frame.

3.2.2.8. Dropping Data Sets with No Sentiment Value

After non-feeling words were excluded from the *HtagSenti* data frame, the contents of the *Allinfo* data frame need to be synchronized with the changes as well. The *Allinfo* data frame was now containing five columns, *ID*, *image_URL*, *hashtag*, *preprocessed_htag* and *mark*. We identified the hashtags with non-zero sentiment value in the *preprocessed_htag* column by comparing the contents with the *HtagSenti* data frame and added a new column, *htag_with_senti*, to the *Allinfo* data frame for storing the extracted results.

In the *htag_with_senti* column, there were several empty values, which means that some data points did not contain any feeling words (hashtags); we then dropped out the rows of these data points. The number of data sets stored in *Allinfo* became 5,240, and these sets of data were the final data remained to be analyzed.

3.2.3. Image Segmentation

In the encrypted image file, there were still 5,397 images. We recorded the IDs of the dropped rows in the previous step and excluded the images with the same IDs through the Jupyter Notebook, and thus the final number of images was 5,240 as well.

In order to protect the privacy of the image owners and make the colour observation more intuitive during the visualization, we pixelated the images into colour segments. The technique we used to pixelate images was the Python module, scikit-image, a collection of algorithms for image processing.

The *slic()* function from scikit-image helped us segment the image into small areas according to the colour properties of the pixels. Each of our candidate images were comprised of 47,089 pixels [8], and each pixel consisted of three colour channels, R (Red), G (Green) and B (Blue). When loading images into the Jupyter Notebook, each of them was transformed into a number array with 217 rows and 217 columns that contained the colour information of the 47,089 pixels. The *slic()* function converted all (R, G, B) values of an image into (X, Y, Z) values defined in the CIEXYZ colour system[9], and created a five-dimensional feature vector to present each pixel's location coordinates and its colour properties as (x, y, X, Y, Z). The function then used K-Means method to cluster the 47,089 feature vectors of an image, classifying pixels with similar properties into the same segment; eventually, pixels within the same segments were assigned the same labels, and an 217*217

[8] All our candidate images were formatted into the same size with 217 pixels in width and 217 pixels in length.

[9] A quantitative defined colour description system considering the mathematical transformation between physiologically perceived colours in human vision and distributions of wavelengths in the visible electromagnetic spectrum.

array containing the label information was returned.

We tuned the parameter to make *slic()* function separate every image into sixty segment areas, which could highly pixelate the images for anonymity while still provide enough colour variety. We then calculated the mean (R, G, B) values of pixels with the same labels and re-assigned these new average colours to each segment. After the segmentation, every image contained up to sixty different colours with distorted contours barely revealing the original scenes. At the end of this stage, the 5,240 segmented images were stored into another encrypted file to be analyzed.

3.2.4. Creating Feature Vectors for Visualisation and Clustering

3.2.4.1. Selecting Clustering Method: Hierarchical Clustering

Our goal of clustering was to classify the images by their sentiment and semantic properties and find the colour patterns across classified groups. The sentiment properties could be determined by the opinion mining tools, while it was hard to define the semantic relation among the hashtags used in social media posts; hence, an automatic algorithm was required to help with the classification. Since we didn't have the ground truth labels for the existing data [17], an unsupervised clustering technique was chosen.

Hierarchical clustering, also known as dendrogram, and K-Means were the two most popular methods for unsupervised clustering. Generally, the results of K-Means had better interpretability when features within a cluster had roughly equal variance; moreover, it often produced clusters with relatively uniform size, and thus the result was sensitive to outliers. In accordance with the preview of the 3D

t-SNE visualisation (**Figure 2**), our data was not evenly distributed in the Euclidean space and did include several outliers, so the K-Means method was not the prior selection for us to conduct clustering. On the contrary, Hierarchical clustering could deal with clusters of varying size and density, having relatively low sensitivity to noise and outliers; furthermore, it did not randomly select start points, which avoided the instability of results due to different initial random states [18]. Hence, Hierarchical clustering was chosen for our project.

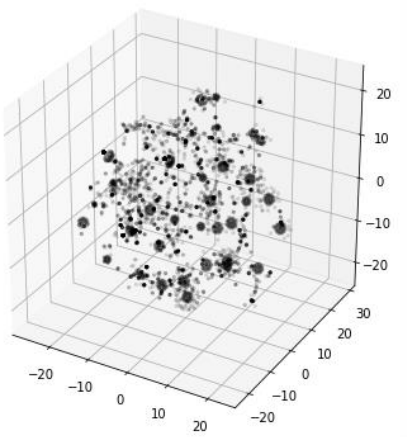


Figure 2: Preview of the sixteen-dimensional data in 3D space

3.2.4.2. Feature Vector: sixteen initial hashtags

We used the selected sixteen initial hashtags to form the features of each image. In the *Allinfo* data frame, we added a new column, *feature_vector*, to store the sixteen-dimensional list that contained the counts of each initial hashtag appearing in the *preprocessed-htag* column of every data point (see **Figure 3**); the names of each vector item were, orderly, [*love*, *beautiful*, *smile*, *amazing*, *inspiration*, *delicious*, *happiness*, *relax*, *sad*, *cry*, *depressed*, *upset*, *hate*, *tears*, *stressed*, *heartbroken*]. Since we had done the word lemmatization, some of the initial hashtags were converted into different grammatical type, such as “amazing” to “amaze”; furthermore, when sifting through the *HtagSenti* data frame, we found

that there were several similar words with only grammatical difference which could not be uniformed by the lemmatization tool, such as “stress” and “stressful”. We took these situations into account when forming the vectors, and when counting the hashtag (1) *amazing*, “amaze” was counted instead, (2) *inspiration*, “inspire” and “inspirational” were counted as well, (3) *depressed*, “depress”, “depression” and “depressive” were counted, (4) *tears*, “tear” was counted instead, (5) stressed, “stress” and “stressful” were counted. Hence, if a data point had a pre-processed hashtag list [love, love, sad, stress, tear, depress, depressive], the sixteen-dimensional feature vector of that data point would be like *[2, 0, 0, 0, 0, 0, 0, 0, 1, 0, 2, 0, 0, 1, 1, 0]*.

ID	*image_URL*	*hashtag*	*preprocessed_htag*	*mark*	*feature_vector*

Figure 3: Columns of *Allinfo* data frame

3.2.4.3. Feature Vector: sentiment feature

The feature vectors created through the sixteen initial hashtags did not reveal sentiment properties; thus, if we used the technique that cluster data according to the Euclidean distance between points, the data would be classified merely by the frequency of the appearance of initial hashtags while the positive and negative features were ignored. For example, the vectors *[1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]* and *[0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]* contained only positive hashtags and *[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1]* contained only negative hashtags; however, as these three vectors had the same Euclidean distance between each other, they were highly likely to be clustered into the same group. Taking sentiment properties into consideration, we added another three sentiment features to form the feature vectors; the names of these features were *all_positive*,

all_negative and *both_senti*. *All_positive* referred to that, in the *htag-with-senti* column of *Allinfo* data frame, the data point included only hashtags with positive sentiment value. *All_negative* worked similarly to *all_positive*, but with negative sentiment value only.

Studies had shown that the human brain operated differently when processing the semantic integration of bivalent constituents, such as "*abused (-) child (+)*" [19]. In our research, there were several similar cases, such as *love* and *upset* existing in the same Instagram post; we considered this situation as a property and gave a value to the feature *both_senti* if a data point included both positive and negative hashtags in the *htag-with-senti* column of *Allinfo* data frame.

For convenience, we took [*all_positive, all_negative, both_senti*] as the partial feature vector of each data point to explain the value assignment. (1) If the data point contained only positive hashtags, it would be assigned [3, 0, 0] , (2) if the data point contained only negative hashtags, it would be assigned [0, 3, 0], (3) if the data point contained both positive and negative hashtags, it would be assigned [0, 0, b_i], in which b_i was defined as:

$$b_i = 1 + \left(\frac{n_i}{n_i + p_i}\right) \times 2 \quad (1)$$

In Equation (1), n_i means that for data with *ID i*, the number of negative hashtags in the *htag-with-senti* column of *Allinfo* data frame; p_i means that for data with *ID i*, the number of positive hashtags in the *htag-with-senti* column of *Allinfo* data frame.

- **Why choosing 3 as the limit of given values in sentiment features**

In the result of Hierarchical clustering for the original sixteen-dimensional feature

vectors, the average centroid distance between the final clusters was about 3 (2.96). In order to differentiate the sentiment properties while still maintain the original semantic relationship of words, we restricted the assigned value of each sentiment feature to be bigger than 1 and no more than 3. Eventually, the value 3 was chosen due to the higher average silhouette score of clustering result.

The constructed nineteen-dimensional feature vectors were then stored in the *feature_vector* column of *Allinfo* data frame. However, the vectors were now stored in "list" format; to make them suitable for the following processing stages, we extracted elements in these lists and created a data frame, *feature_vector_df*, to store them in different columns. The index of *feature_vector_df* data frame was the data ID and the feature names of the feature vectors were the names of nineteen columns.

3.3. Data Visualisation for Exploration

3.3.1. Dimensionality Reduction

The challenge of conducting the high-dimensional data visualization was to figure out general representations of data that could display the multivariate structure of several variables at a time and in an accessible two-dimensional or three-dimensional space [20].

To suit our nineteen-dimensional data into lower dimensional space, a dimensionality reduction process was required, and two popular methods were PCA (Principal Component Analysis) and t-SNE (t-distributed stochastic neighbor embedding). PCA is a statistical technique which constructs a linear

transformation to extract the most important features that capture the maximum information (cause highest variance) about the dataset and uses the extracted principal components in the analysis instead of using the original features. However, due to its linearity, PCA does not attempt to group similar points together and cannot well interpret the relationship of data points according to their relative distance in the low dimensional space [21].

T-SNE, on the other hand, is a nonlinear dimensionality reduction algorithm which allows users to separate data that cannot be separated by any straight line. It starts from creating a Gaussian probability distribution representing similarities [10] between neighbors based on the original features, and eventually find a similar Student t-distribution in a low-dimensional space [22]. Owing to its ability to group data with similar properties, t-SNE was selected to be the dimension reduction technique for our data visualization.

3.3.2. Embedding Projector

The general two types of data displays were the dynamic type, which allowed user interaction, and the static one; it turned out that the interaction with the visual display was crucial in order to gain a better insight to the data during the high-dimensional data exploration [20].

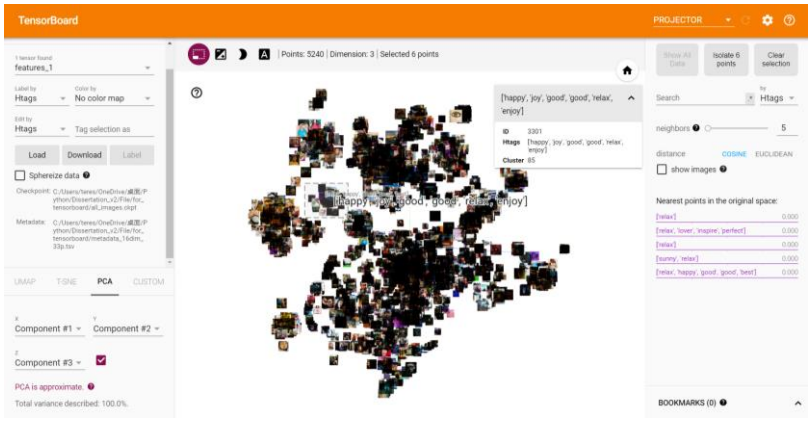
The Embedding Projector was a web application integrated into the TensorFlow platform, and was an interactive visualisation tool that automatically projected high-dimensional data into low-dimensional space by t-SNE or PCA method, allowing users to view the whole embedding process dynamically. However, since

[10] The similarity of datapoint x_j to datapoint x_i is the conditional probability p_{j|i}, that x_i would pick x_j as its neighbor.

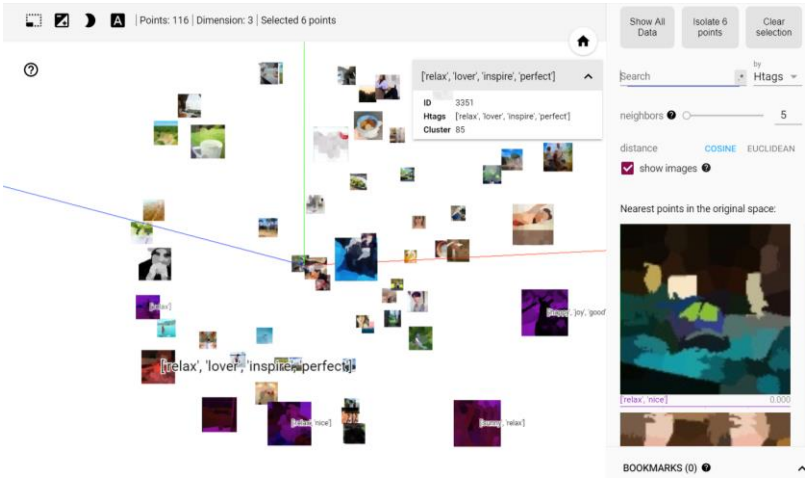
monitoring the embedding process was not our main goal, we directly uploaded the t-SNE preprocessed data into the Embedding Projector and used three-dimensional PCA mode to view the result.

To initial the Embedding Projector, we need to prepare the necessary information for uploading. Firstly, we created a TSV file to record our nineteen-dimensional feature vectors in tab-separated format without index and column names, and used the *tensorflow.Variable()* function to read and store these data as the embedding variables. Secondly, we created another TSV file for storing the supporting information, such as the label, category or specific properties of each data point, and this information could then be displayed on the working panel of the Embedding Projector. Thirdly, in order to replace all data points by their corresponding images during visualisation for colour observation, we created a "sprite image" for TensorFlow to identify the data-image pairs. A sprite image was a square canvas with a collection of images neatly aligned on it, where the first data point placed in the top left and the last data point placed in the bottom right; the TensorFlow could then link the images and feature vectors orderly. Finally, we used the *tensorflow.Session()* function to initialize a session that saved the embedding variables and check points into a specified logging directory; the sprite image and the TSV file of the supporting information were saved into the same directory as well. After running a command in the Python Terminal window, we could view the visualisation result in the browser (**Figure 4(a)**).

Users can zoom, rotate and select data using natural click-and-drag gestures, viewing the information of each data point, such as the label, category or other user-defined content, and isolate the points with similar properties to the selected one (**Figure 4(b)**).



(a)



(b)

Figure 4 (a)(b): Data visualisation in the Embedding Projector
(a) shows the visualisation of t-SNE processed image data. (b) shows the isolated points of the selected compact cluster in (a) and the magnified images on the right panel.

3.3.3. Implementation in the Unity

The other tool we used for visualisation was the Unity, a cross-platform game engine giving users the ability to create experiences in both two-dimensional and three-dimensional environments. In the Unity, we modelled a 3D scatter plot displaying the data points according to their coordinates produced by t-SNE technique and replaced each data point by its corresponding image. It was noteworthy that the images were presented in cubes because we found that, when

visualising image data in the Embedding Projector, many images became translucent to show the ones overlapped behind, which made it hard for us to observe the colour patterns. To improve visualisation performance, we turned images into cubes with the same images on six sides; by doing this, the covered images could be seen from the other sides, which also made it more convenient for observation from any perspectives when we navigated through the data points.

3.3.3.1. Identification of Observed Clusters

During the visualisation, we found that some data points gathered as a compact sphere, some closely linked to each other as a line, while some were just shapelessly scattered around. The conspicuous colour patterns we observed were mostly from clusters in spheres and lines.

To specify the observed clusters, we used the Hierarchical clustering algorithm to classify the data points into 80, 100 and 120 clusters, respectively, to compare the effectiveness of these three parameters. We then painted each cluster of data with different colours (with the help of Embedding Projector) so that we could see how the algorithm worked (**Figure 5**). From the three visualisation results, we chose the one whose classification was the closest to that of our observation; eventually, the one with 100 clusters was chosen.

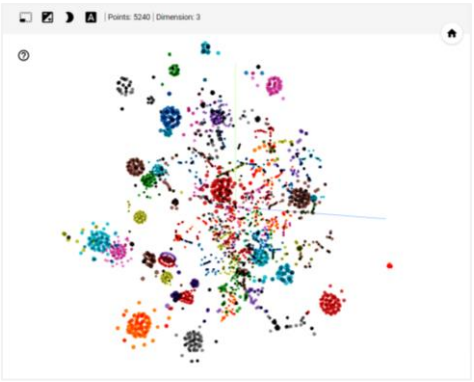
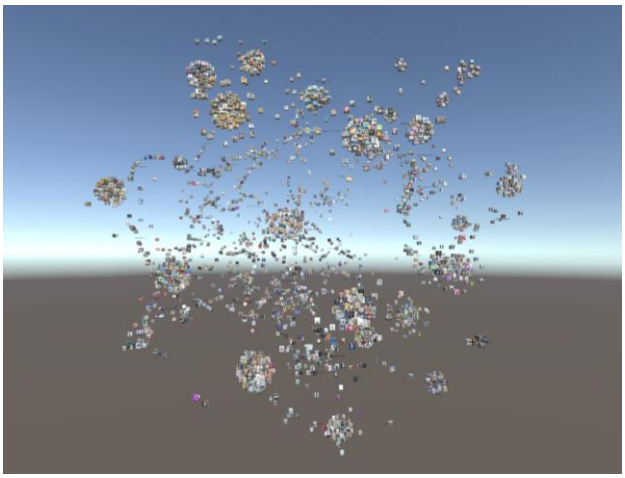


Figure 5: Clusters painted with different colours in the Embedding Projector

In order to identify the observed groups and explore the semantic information, we generated the sequential numbering, from 1 to 100, and representative hashtag labels for each of the one-hundred clusters; the label showed the four most frequently used hashtags of each cluster. With the operation of the keyboard and mouse, we were able to navigate through the 3D scatter plot on the computer by going up, down, left, right, forwards, backwards and from different perspectives (**Figure 6 (a)**). Furthermore, when implementing the visualisation in the Data Visualisation Observatory of the University of Manchester, the researchers were able to immerse themselves in the data, walking around and explore the data in the best possible vision (**Figure 6 (b)**).



(a)

(b)

Figure 6 (a)(b): Data visualisation in the Unity
(a) shows the visualisation of t-SNE processed image data.
(b) shows the implementation of visualisation in the Data Visualisation Observatory of the University of Manchester

3.4. Data Mining Task I: Data Clustering

3.4.1. Hyperparameter tuning

We aimed to classify many data points into fewer number of clusters; following the same ideas, we used the Agglomerative clustering, a bottom-up approach of Hierarchical clustering that treated each data as a singleton cluster at the beginning and then successively agglomerated pairs of clusters until all clusters being merged into a global cluster containing all data. The tool we used for clustering was the *AgglomerativeClustering()* function in Python scikit-learn module.

Typically, Hierarchical clustering does not require a prespecified number of clusters; however, in our application, we wanted a partition of disjoint clusters, such as in flat clustering[11], to conduct colour analysis on different groups. In addition, to reach better and suitable clustering results, we also need to specify the

[11] Flat clustering creates a flat set of clusters without any explicit structure that relates clusters to each other and requires scientists to tell the machine how many categories to cluster the data into.

way the algorithm calculated the distance between clusters. Hence, there were two essential hyperparameters to be tuned in the *AgglomerativeClustering()* function, which were the prespecified number of clusters, *n_clusters*, and the cluster distance calculation criterion, *linkage*.

3.4.2. Tuning *n_clusters*: Silhouette method

The way we used to determine the optimal number of clusters was the Silhouette method. The silhouette score measured how similar a data point is to its own cluster compared to the other cluster by using the Equation (4) [23]. There were two variables in the silhouette score equation; *a(i)* represents the mean distance between the data point *i* and all other data points in the same cluster, and *b(i)* represents the smallest mean distance of the data point *i* to all points in any other clusters.

In the equation calculating *a(i)* (Equation (2)), C_i refers to the cluster which data point *i* lied in and *d* refers to the Euclidean distance between data point *i* and *j*. In another equation calculating *b(i)* (Equation (3)), C_k refers to all other clusters except for C_i and *d* refers to the Euclidean distance between data point *i* and *j*.

The Silhouette method would calculate a score *s(i)*, with the given *n_clusters* (the number of prespecified clusters), by dividing the difference between *b(i)* and *a(i)* by the larger value of either of these two variables.

$$a(i) = \frac{1}{|C_i| - 1} \times \sum_{j \in C_i, i \neq j} d(i,j) \tag{2}$$

$$b(i) = min_{k \neq i} \frac{1}{|C_i|} \times \sum_{j \in C_k} d(i,j) \tag{3}$$

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}, if\ |C_i| > 1 \tag{4}$$

Lastly, we calculated the average silhouette score of all data points at each given *n_clusters* state and figured out the optimal number of clusters that gave the highest average silhouette score. To do this, we created a loop and run iterations by tuning the number of clusters from 2 to 70 and plotted a line chart showing the variation of silhouette scores (**Figure 7**). The result showed that when *n_clusters* equaled 5, the clustering model could have the highest average silhouette score at around 0.453. Although the line seems to gradually bounce up after *n_clusters* equaled 20, the average silhouette score still performed much worse even until seventy clusters were generated; therefore, we selected 5 as the number of clusters to be classified into. **Figure 8** illustrates the silhouette score of each data point and shows the cluster size when *n_clusters* equals five. The resulting five groups of data points were named as *Group0*, *Group1*, *Group2*, *Group3* and *Group4* for further analysis.

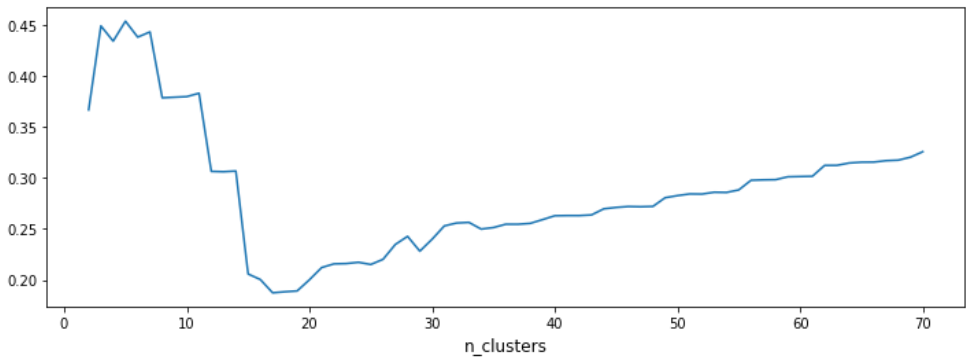


Figure 7: The average silhouette score at given n_clusters

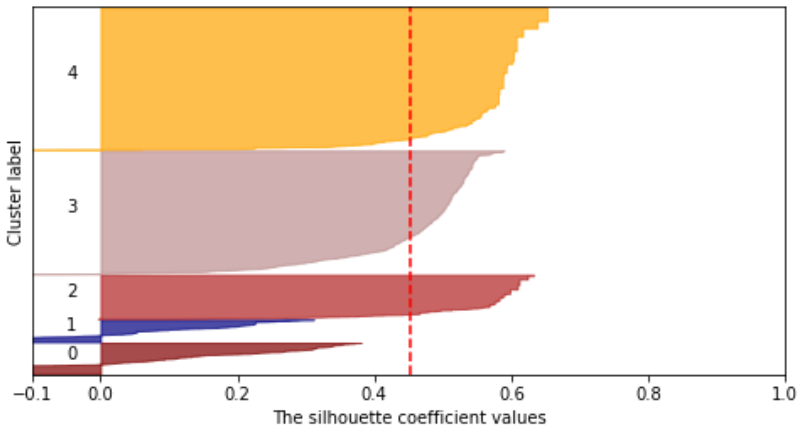


Figure 8: Silhouette analysis with n_clusters = 5

3.4.3. Tuning *linkage*

When calculating the distance between data points of different clusters, there were four candidate methods in the *AgglomerativeClustering()* function, which were "single", "complete", "average" and "ward". Single linkage deemed the proximity of two clusters as the distance between their two closest data points, while complete linkage deemed the proximity of two clusters as the distance between their two farthest data points. Average linkage considered not only two points but the average of all pairwise distance between two clusters. Ward linkage, with more complexity, assumed that each existing cluster had its own centroid and its own variance, which was the sum of square of the distance between the centroid and other data points in the same cluster; the ward linkage then defined the proximity of two clusters as how much the variance would increase after they were merged together [13].

Among the four linkage methods, the single linkage controlled only nearest neighbor similarity and could easily cause bias when two very dissimilar clusters happened to have two very similar points; complete linkage could generate clusters with compact contours by their borders while not necessary compact inside;

average linkage and ward linkage could avoid the problems of the other two methods above, while the former was suitable for clusters of miscellaneous shapes and the latter aimed at finding compact, spherical clusters (**Figure 9**).

From the previous data visualisation, we could see that a large amount of our data gathered as compact groups; since we would like to use a method that clustered data along the way we observed, which means to have a relatively compact center in each resulting cluster, the ward linkage method was chosen.

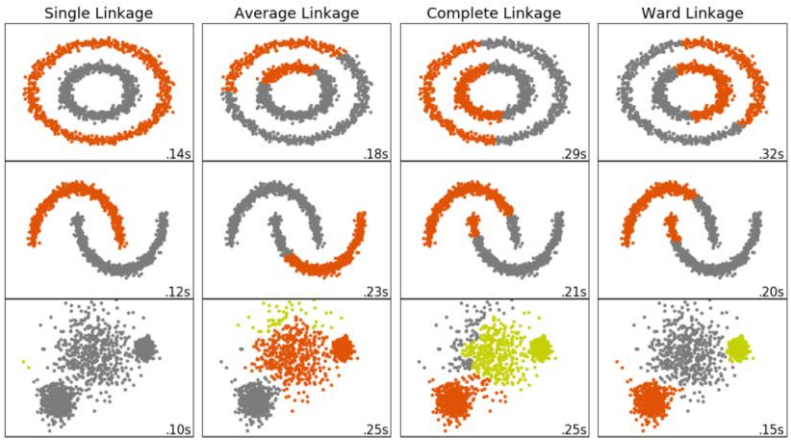


Figure 9: Comparison of different hierarchical linkage methods

3.5. Data Mining Task II: Colour Analysis

3.5.1. Introduction of Colour System

A colour model is a mathematical system that represents visible colours (perceivable by human eyes) formally in a tuple of numbers, typically as three or four colour components depending on the model used. When the model is well-defined with the way the colours are to be interpreted, such as the quantitative viewing conditions, the resulting set of colours forms the corresponding colour space. Different colour models can be converted to the form of each other through

mathematical equation, while they are still divided into the following three categories according to image processing applications [24]. (1) Device-dependent colour models: the resulted colours are affected by the tools used for displaying, such as RGB and CMY(K) models, (2) User-oriented colour models: considered as the communication bridge of colour information between the observer and the displaying device, and enables human to objectively describe what they percept from the presented colours; the models of this class are such as HSV and HSL, (3) Device-independent colour models: the resulted colours are not affected by the device properties, and the same colour is determined by a fixed set of parameters during the transmission through different hardware devices; the representative model is the Munsell colour model.

We aimed to find out how human's usage of colours related to their moods and would also evaluate the image data visualisation results through direct observation, and thus the user-oriented colour models which operate more similarly to the way our eyes and brain interpret colour is preferable.

Among user-oriented models, the HSV (Hue, Saturation, Value) colour space was chosen for our project since the definition of "saturation" in HSL model is controversial; for example, a very pastel and nearly white colour can be described as fully saturated in HSL, which is contradictory to the definition of saturation "the colourfulness[12] of an area judged in proportion to its brightness" [25].

The HSV colour space could be visualised as a cone object with the hue varying along the outer circumference of a cone, the saturation increasing with distance

[12] Colourfulness is the attribute of a visual perception according to which the perceived color of an area appears to be more or less

from the centre of a circular cross-section, and the value increasing from bottom to top (**Figure 10**) [26]. To avoid confusion, the *value* of colour would written in italic in the following contents. In the HSV system, the hue refers to the visual attribute according to which the area seems to be like any of the perceived colours, red, green, blue, or a combination of them; the saturation, as defined above, refers to the colourfulness relative to the "white" at the same illumination level; the *value* is the brightness of the colour relative to the brightness of "white" at the same illumination level.

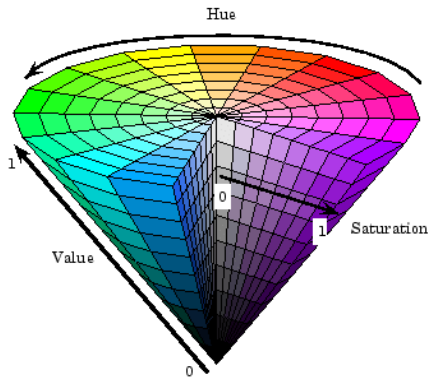


Figure 10: HSV colour solid cone[13]

3.5.2. Analysis Process

During the colour processing stage, the tool we used to extract colour and transform colour systems was OpenCV, a library of programming functions mainly aimed at real-time computer vision processing, with the ability to process multiple vision/image related tasks and was binding in Python as the module *cv2*.

The images were originally stored in RGB colour format when loaded into the

[13] Image source: *Poorani, M., T. Prathiba, and G. Ravindran, Integrated feature extraction for image retrieval. International Journal of Computer Science and Mobile Computing, 2013. 2(2): p. 28-35.*

Jupyter Notebook, and thus we used the *cvtColor()* function in the module *cv2* to convert the RGB value of the pixels in each image into HSV format for analysis. The conversion relationship between the RGB and HSV colour system can be found in [27].

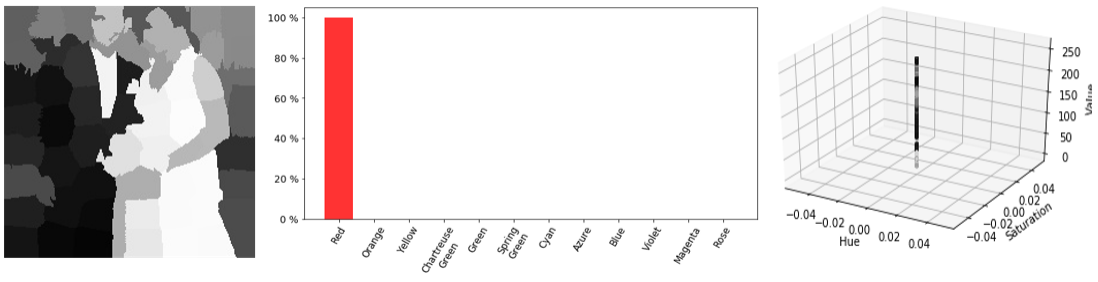
We created an HSV data frame of hue (*h*), saturation (*s*) and *value* (*v*) for each of the five Hierarchical clustered groups. Take *Group0* as an example, the index of its HSV data frame were the *ID* of data points within the group; for data with *ID 0*, the *h, s, v* columns stored a list of 47,089 elements, respectively, describing the colour information of every pixel in the image with the same *ID*.

3.5.2.1. Hue

Practically, the value of hue is measured in degrees of the color circle (hue spectrum) ranging from 0° to 360°, where 0° refers to red, 120° is green and 240° is blue. To describe colours more generally, we divided the hue spectrum into twelve sections of which the representative colours were named as Red ($346^{\circ}\sim360^{\circ}, 0^{\circ}\sim15^{\circ}$), Orange ($16^{\circ}\sim45^{\circ}$), Yellow ($46^{\circ}\sim75^{\circ}$), Chartreuse Green ($76^{\circ}\sim105^{\circ}$), Green ($106^{\circ}\sim135^{\circ}$), Spring Green ($136^{\circ}\sim165^{\circ}$), Cyan ($166^{\circ}\sim195^{\circ}$), Azure ($196^{\circ}\sim225^{\circ}$), Blue ($226^{\circ}\sim255^{\circ}$), Violet ($256^{\circ}\sim285^{\circ}$), Magenta ($286^{\circ}\sim315^{\circ}$) and Rose ($316^{\circ}\sim345^{\circ}$). In the Python OpenCV module, the range of hue was cut half to be 0 to 180 (without degree units) for calculation convenience, and thus the range of each representative colour defined above was also cut half during our programming.

To figure out the hue pattern of the five groups, we transformed the whole *h* column of the data frame into a flattened array and calculated the frequency of values in the array that lied within each hue interval of the twelve representative

colours. It was found that some results showed high frequency of Red colour even in the image that barely had reddish colour in it (**Figure 11 (a)(b)**). The reason was that, practically, the hue of pixels without saturation was not detectable; however, when processing the colour of black-and-white images (images without saturation), the function in Python OpenCV module assumed the hue value of all pixels to be 0 (**Figure 11 (c)**), which happened to lie in the hue interval of Red. To avoid misleading, we did not consider the black-and-white images when demonstrating the hue performance of in the bar chart.



(a) (b) (c)

Figure 11(a) – (c): Colour performance of black-and-white image

Furthermore, we attained a representative hue of each group by calculating the average hue value. Since the hue was quantified in circular degrees, the general arithmetic method might not give an appropriate average value. For example, the arithmetic means of 0° and 360° is 180°, which is misleading because for most purposes 360° could be seen as 0°. Hence, we used the Mean of Circular Quantities method [28] to calculate the representative hue by converting polar coordinates to Cartesian coordinates and computing the arithmetic mean of the converted points; finally, the results were transformed back into polar coordinates, and the corresponding angle ($\bar{\alpha}$) was the reasonable mean of the input angles (α_i) (Equation (5)).

$$\bar{\alpha} = atan2\left(\frac{1}{n}\sum_{i=1}^{n}\sin\alpha_i\,,\frac{1}{n}\sum_{i=1}^{n}\cos\alpha_i\right) \quad (5)$$

3.5.2.2. Saturation

In the practical HSV colour system, the value of saturation ranges from 0 to 1, while the OpenCV module extends the range to from 0 to 255. When investigating the saturation performance of images within a hierarchically clustered group, we calculated the average saturation of each image and constructed a plot to show the distribution of these averages, with x-axis being the saturation value and y-axis being the frequency. We then computed the mean value and the ninety-five percent confidence interval of these averages to gain insight into the group's overall saturation properties. During this process, an array with 217 rows and 217 columns was created for each image data point to store the saturation value of all pixels; after generating the mean value of every array, we created a list to store the results for statistical demonstration. However, it was found that there was a high frequency of the value 0 showing in the previewed histogram of hue, which means several images were without saturation, and this could happen on black-and-white images. To specify this property, we separated black-and-white images from the others when demonstrating the results in charts.

3.5.2.3. Value

In the practical HSV colour system, the value of *value* also ranges from 0 to 1, while the OpenCV module extends the range to from 0 to 255. When investigating the *value* performance of images within each group, we calculated the average *value* of each image and constructed a plot to show the distribution of these

averages, with the x-axis being the value of *value* and y-axis being the frequency. We then computed the mean and the ninety-five percent confidence interval of these averages to gain insights into the group's overall *value* properties. During this process, an array with 217 rows and 217 columns was created for each image data point to store the *value* information. After generating the mean of every array, a list was created to store the results for statistical demonstration.

Section 4: Data Analysis and Findings

4.1. Hierarchical Clustering Results: hashtag property

The feature vectors we used for classification were formed mainly based on the sixteen initial hashtags. To gain insights into the properties of the resulting five groups, we counted the average proportion of the initial hashtags in all emotional hashtags of each data point in the group. For example, if a data point had the hashtag list *[love, love, happiness, beautiful, cute]* in the *htag-with-senti* column of the *Allinfo* data frame, its proportion of *love* is given by (2/5)*100%; we then calculated this proportion for all data points within a group and computed an average value. This process was repeated for all the other initial hashtags, and the results of each group were demonstrated through the bar charts for further interpretation.

4.1.1. Group0

There were 450 data points in *Group0*. In **Figure 12(a)**, the bar chart shows that data points in *Group0* contained both positive and negative sentiment. However, almost all positive sentiments came from *love*, accounting for just slightly less than forty percent, while the other positive hashtags were rarely used in the group. On the other hand, the negative part was dominated by *sad*, *hate* and *heartbroken*, which accounted for about ten percent, two-point-five percent and five percent, respectively.

4.1.2. Group1

There were 325 data points in *Group1*. In **Figure 12(b)**, the bar chart of *Group1* also shows a combination of positive and negative sentiment. The positive hashtags used were dominated by *love*, but with only about seven percent, and the other positive hashtags were barely used. As regards negative hashtags, *sad* was the most frequently used one, accounting for an average of more than twenty-five percent, while the others only made up two to six percent.

4.1.3. Group2

There were 637 data points in *Group2*. In **Figure 12(c)**, the bar chart reveals a one-sided result showing that *Group2* contained only hashtags with negative sentiment. Among the negative initial hashtags in *Group2*, *sad*, *cry*, *tear* and *stress* were relatively more often in use than the others, where *sad* constituted the largest proportion at about seventeen percent, *cry* and *tear* accounting for about thirteen percent, respectively, and *stress* made up slightly less than ten percent. The rest of the initial negative hashtags accounted for six percent or less.

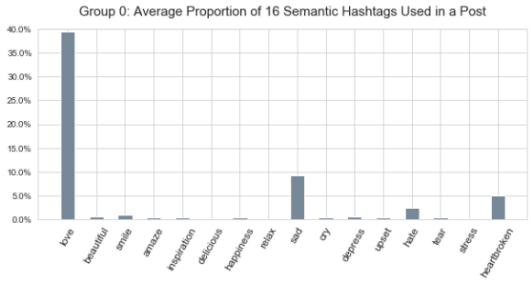
4.1.4. Group3

There were 1781 data points in *Group3*. In **Figure 12(d)**, the bar chart shows a more balanced distribution of positive and negative hashtags appearing in *Group3* compared with in the other groups. However, *love* constituted an average of about eleven percent of hashtags used, which is much higher than the others did. The usage rate of *sad*, *cry*, *hate* and *tear* ranges from about three percent to no more than five percent, and the rest of the initial hashtags accounted for less than two percent, respectively.

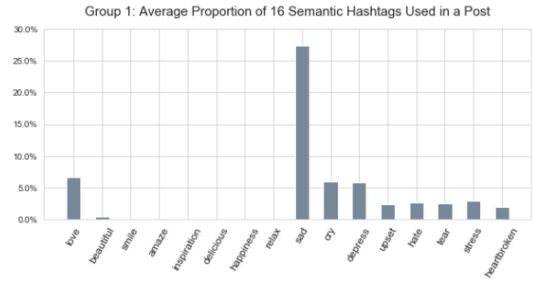
4.1.5. Group4

There were 2047 data points in *Group4*. In **Figure 12(e)**, the bar chart reveals a one-sided result showing that *Group4* contained only hashtags with positive sentiment. Among the positive initial hashtags used in Group4, *love* and *relax* had relatively higher usage rate than the others, where *love* accounted for the largest proportion at about fifteen percent and *relax* constituted more than nine percent. The usage rate of the rest of initial positive hashtags ranges from approximately three percent to six percent.

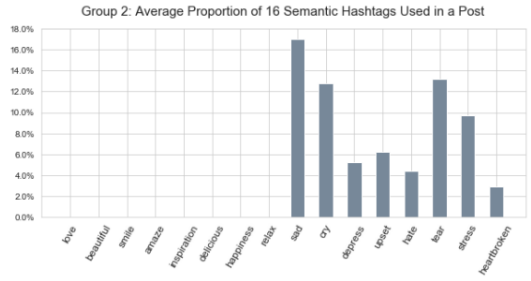
Overall, *Group0*, *Group1* and *Group3* all delivered a mix of both positive and negative emotions. *Group0* featured the dominant hashtag, *love*, and the negative hashtag, *sad*; *Group1*, on the contrary, featured the dominant hashtag, *sad*, and the positive hashtag, *love*. *Group3* had a relatively balanced distribution of all initial hashtags with the relatively dominant one, *love*. *Group2* and *Group4*, on the other hand, contained hashtags with only negative and positive sentiment, respectively.



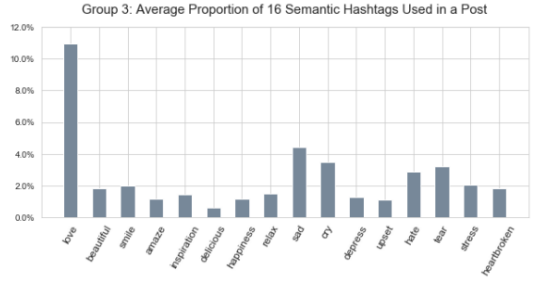
(a)



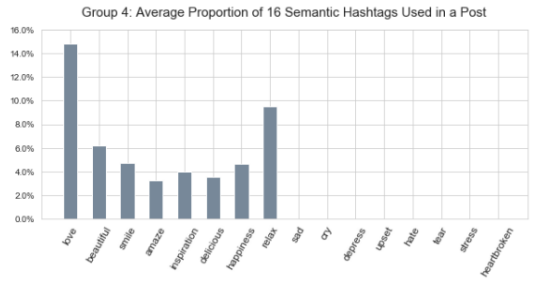
(b)



(c)



(d)



(e)

Figure 12(a) – (e): Average proportion of initial hashtags used in each group

4.2. Hierarchical Clustering Results: HSV analysis

4.2.1. Hue analysis

The bar charts in **Figure 13(a)** to **(e)** illustrate the aggregate proportion of the twelve representative hue colours appearing in images within each group.

The colour Red was frequently used and made up about twenty percent in every group except for *Group4*, which shows a less proportion of Red at 16.03%.

The performance of Orange fluctuated across groups, with the largest proportion in *Group4*, at 35.22%; the second largest proportion in *Group2* and *Group3*, both at about thirty percent, followed by *Group0* at 25.83%, and the least proportion was in *Group1* at 21.81%.

The proportion of colour Yellow did not vary much across groups, all at about ten percent, while the figures were slightly lower in *Group1* and *Group2*, both at about eight-point-five percent.

The three greenish colours, Chartreuse Green, Green and Spring Green, played a small part in the average hue performance; all these three colours constituted no more than five percent, respectively, in each group, while Chartreuse Green

performed slightly better than the others.

As regards Cyan, *Group2*, *Group3* and *Group4* had a proportion of Cyan at around five percent; *Group0* had a higher value at 6.25% while *Group1* had a lower value at 4.02%.

Azure was the third most frequently appearing colour within the groups of image data, after Orange and Red; it accounted for more than ten percent of the hue used in each group, with the largest proportion at 12.94% in *Group1*.

As for Blue, *Group0*, *Group2* and *Group3* had a proportion at around seven percent and *Group4* had a lower value at 5.28%, while *Group1* had the highest value at 10.91%.

The three purplish colours, Violet, Magenta and Rose, also played small roles as the greenish colours did; all of them accounted for no more than five percent, respectively, in each group, while Rose performing slightly better than the others.

In conclusion, the three dominant colours across the five groups were Orange, Red and Azure. After excluding the black-and-white images, Red seemed less dominant while still played a big role in the overall hue performance. *Group1* featured its outstanding performance of Azure and Blue, while it had the smallest proportion of Orange compared with the other groups. *Group4* featured its highest proportion of Orange, while having the least Red compared with the other groups. *Group0*, *Group2* and *Group3* followed an overall pattern of the hue performance with peaks at Red, Orange and Azure; apart from a slightly smaller proportion of Orange in *Group0*, there were barely other distinct features of these three groups.

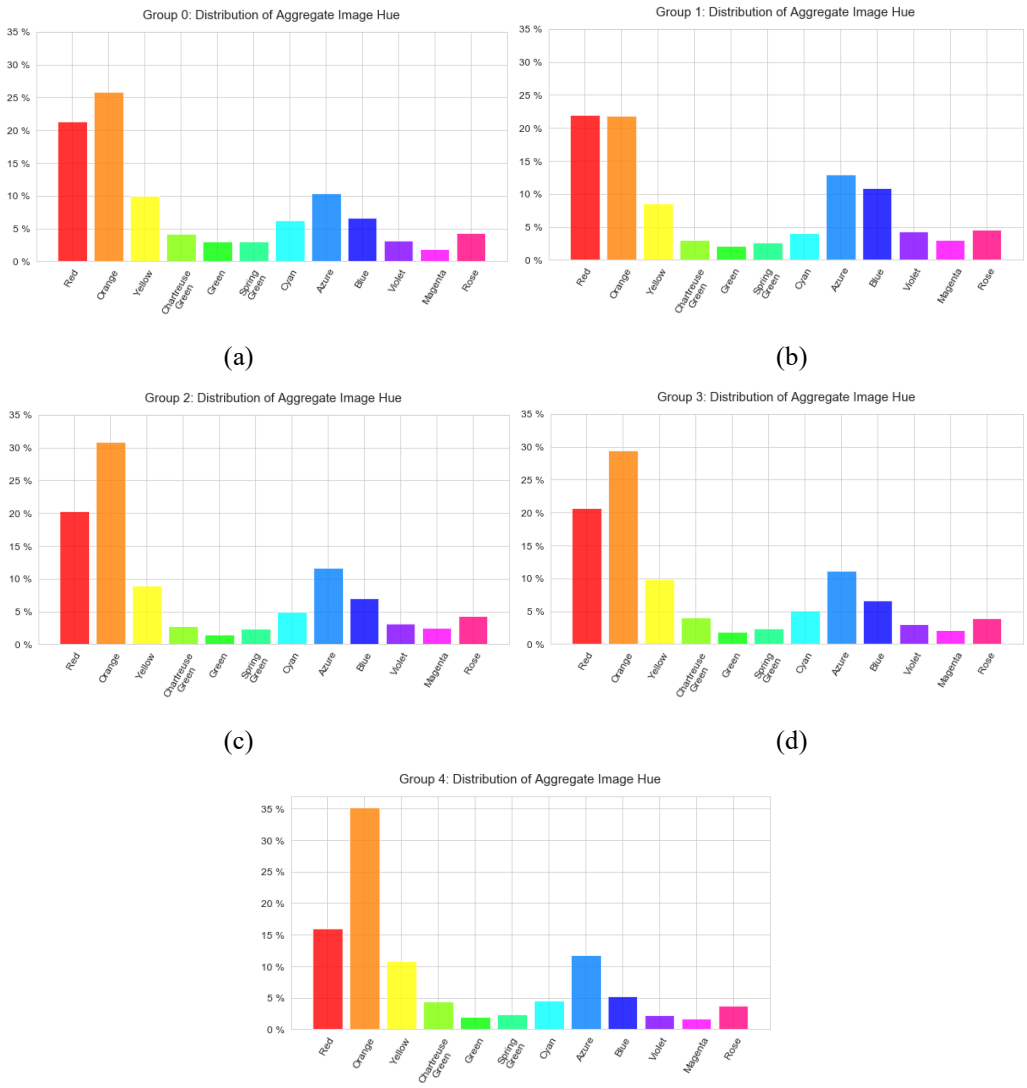


Figure 13(a) – (e): Hue distribution of image data in each group

The Chi-squared test did not prove significant association between the hue and each group, which means images of each group followed a similar colour pattern, as shown in **Figure 14**; however, we could still find subtle properties of certain groups as mentioned above. With regards to the average hue, the Means of Circular Quantities method showed that the average hue of all groups lies in the colour range of Red, which could be the effect of Central Limit Theorem [29]. Considering the previous studies [30, 31] that prove controversial in finding the

average hue perceived through human eyes, we took this result as a reference and left it for discussion in the later section.

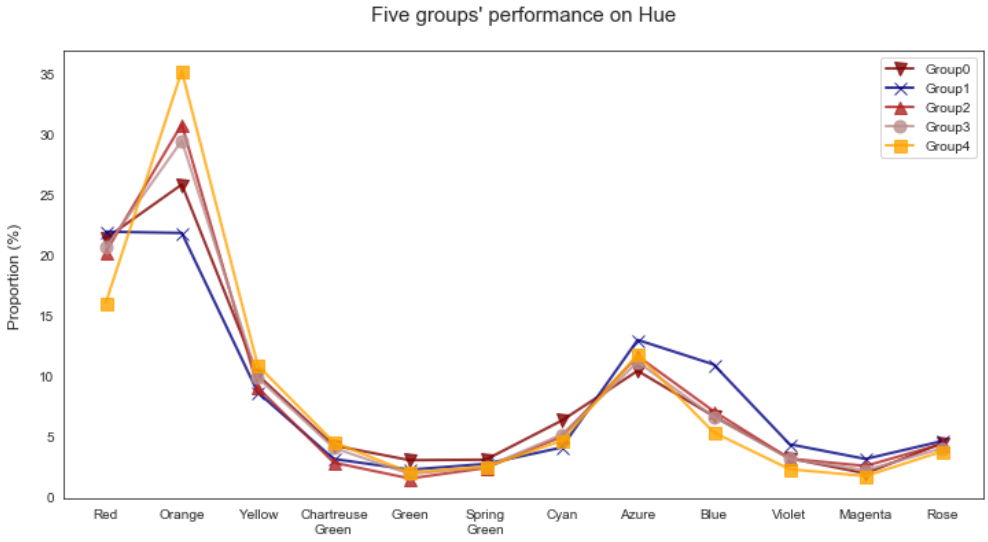


Figure 14: Five groups' performance on Hue
(The colour of lines is only for differentiation)

4.2.2. Saturation analysis

The bar charts in **Figure 15(a)** to **(e)** illustrate the average saturation of images within each group. In the following charts, we demonstrated the results without considering the black-and-white images to gain more accurate insights into the performance of saturation in colourful images.

The number of black-and-white images within each group was 13, 21, 37, 67 and 36, respectively (from *Group0* to *Group4*). However, it is more meaningful to look at their probability of appearance[14], which were 2.9%, 6.5%, 5.8%, 3.8% and 1.8%, respectively. It could be inferred that it is more likely to find black-and-white

[14] The probability of appearance was calculated by dividing the number of black-and-white images within each group by the total number of data points within the same group, formatted into percentage.

images among images with hashtag properties of *Group1*, while for *Group4* it is least likely to be so.

Regarding the colourful images, the mean value and the ninety-five percent confidence interval of the saturation distribution of each group were specified in **Table 1**. According to the table, the order of groups sorted by the mean value of saturation (from high to low) was *Group4*, *Group1*, *Group2*, *Group0* and *Group3*. However, we could only confirm that images with properties of *Group4* had significantly higher saturation than those with properties of *Group3* under the ninety-five percent confidence interval. *Group0*, *Group1* and *Group2* consisted of much smaller number of data points, especially *Group1*, which also had the largest data dispersion. Hence, the resulted confidence intervals of these three groups were much wider, overlapping each other and even covering the range of *Group3* and *Group4*, which made it hard to prove and interpret their saturation performance with solid statistical foundation.

Table 1: Saturation statistics of five groups

	Mean value of Saturation	95% confidence interval	Standard deviation	Number of data points
Group0	75.01	[71.05, 78.97]	42.15	437
Group1	77.39	[71.91, 82.88]	48.59	304
Group2	75.12	[71.53, 78.72]	44.84	600
Group3	74.39	[72.38, 76.40]	42.37	1,714
Group4	78.88	[77.21, 80.56]	38.33	2,011

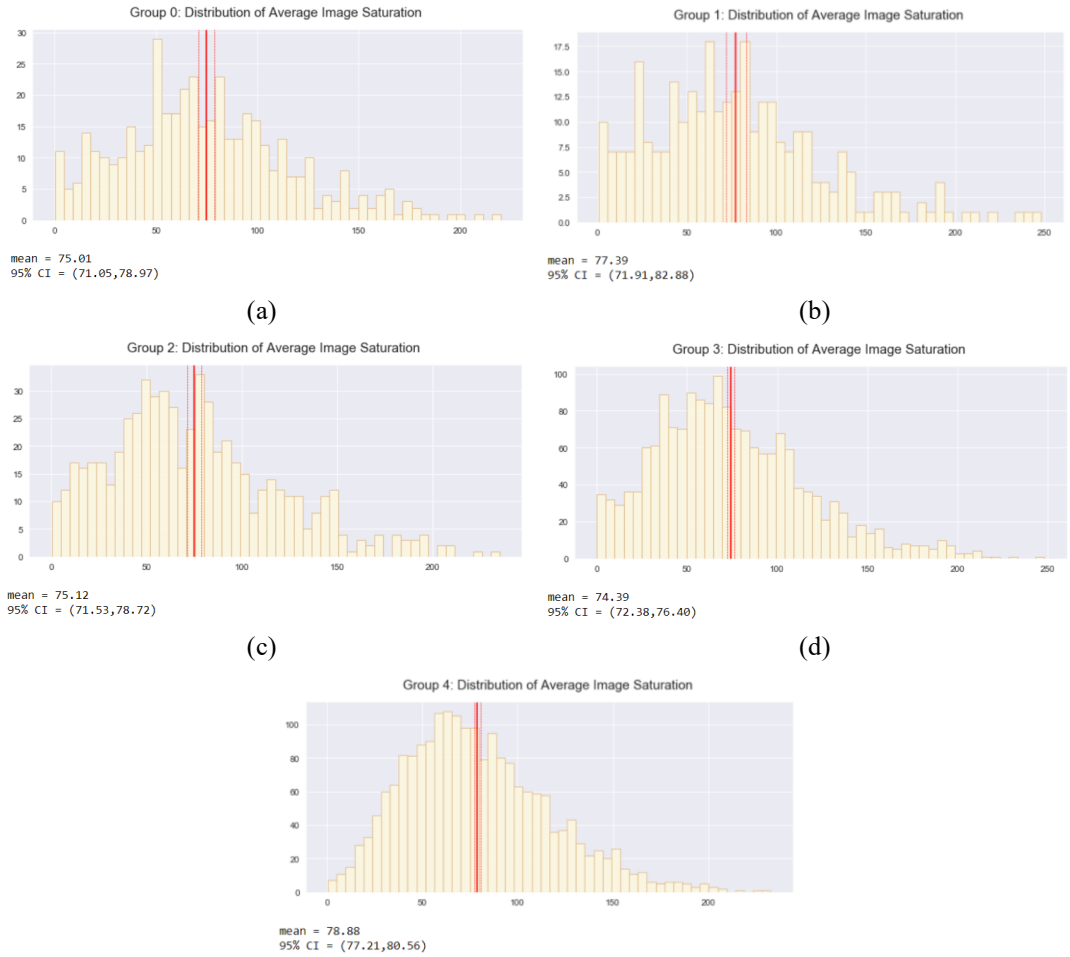


Figure 15(a) - (e): Saturation distribution of image data in each group

4.2.3. Value analysis

The bar charts in **Figure 16(a)** to **(e)** illustrate the average *value* of images within each group. When demonstrating results in the following charts, we did not exclude the black-and-white images as we had done in hue and saturation analysis since images without saturation could still show diversification in terms of *value*.

The mean value and the ninety-five percent confidence interval of the *value* distribution of each group were specified in **Table 2**. According to the table, the order of groups sorted by the mean value of *value* (from high to low) was *Group4*, *Group3*, *Group2*, *Group0* and *Group1*. Within this order, the relationship we

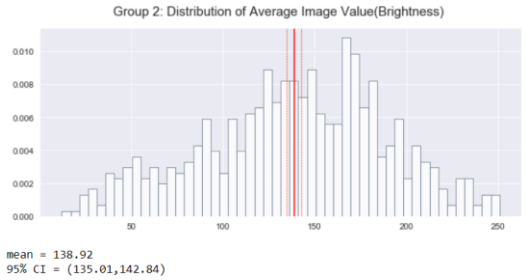
could statistically confirm under the ninety-five percent confidence interval was that the average *value* of images with properties of *Group4* is significantly higher than images with properties of all the other groups; secondly, the average *value* of images with properties of *Group1* is significantly lower than those of *Group2*, *Group3* and *Group4*. It's worth noting that *Group1* had the largest dispersion and smallest number of data points, resulting in the widest range of ninety-five percent confidence interval; however, its confidence interval barely overlapped the others, which proved the much lower image *value* in *Group1* than in other groups.

Table 2: Value statistics of five groups

	Mean value of Value	95% confidence interval	Standard deviation	Number of data points
Group0	132.40	[127.62, 137.18]	51.59	450
Group1	122.90	[117.12, 128.68]	52.97	325
Group2	138.92	[135.01, 142.84]	50.34	637
Group3	139.76	[137.57, 141.95]	47.15	1781
Group4	147.12	[145.45, 148.78]	38.50	2047

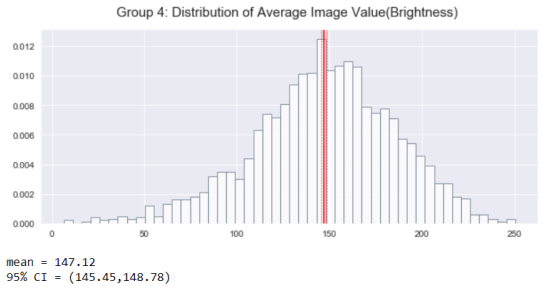
(a)

(b)



(c)

(d)



(e)

Figure 16: Value distribution of image data in each group

4.2.4. Overall Performance

The overall result of HSV colour analysis of each group is shown in **Figure 17** and **Figure 18**. For comparison within our image data, the performance of *Group4* indicates that when all the attached hashtags belong to the eight positive initial hashtags, the images tend to have the highest saturation and *value*. The performance of *Group1* shows that if the images are attached with negative hashtags, among which *sad* accounts for a large proportion, and sometimes with a positive hashtag *love*, the images are likely to have lowest *value* and high saturation. On the other hand, *Group0*, *Group2* and *Group3* show that the saturation is slightly lower and almost with no difference between images attached with only negative initial hashtags and images attached with mixed initial hashtags, if *sad* is not dominant. However, images with mixed initial hashtags where *love* is dominant tend to have lower *value*.

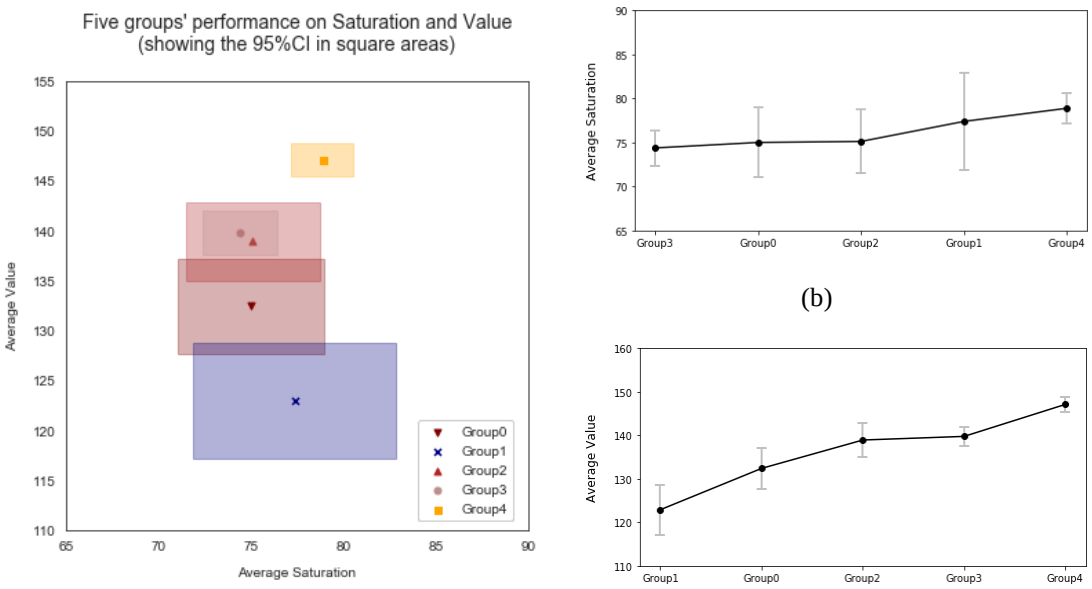


Figure 17(a) – (c): Five groups' performance on saturation and value
(a) The square area shows the co-region of 95% confidence interval of mean saturation and mean *value*. (The colour of squares and points is only for differentiation); (b)(c) show the correlation between saturation/*value* and five groups. (Group orders on x-axis are different in two figures)

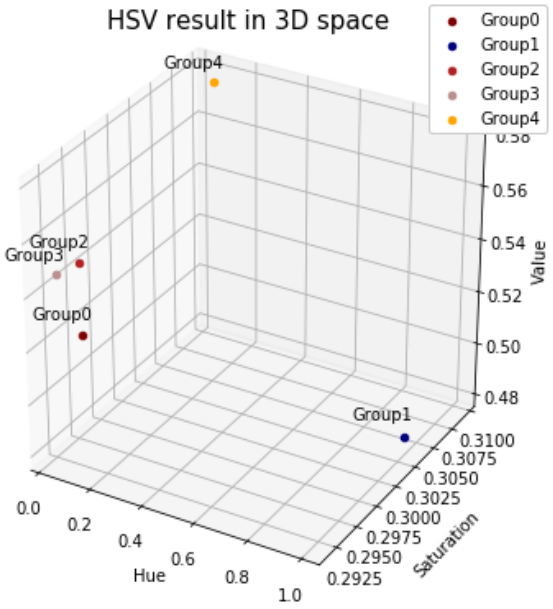


Figure 18: HSV analysis results of five groups in 3D plot
(The colour of points is only for differentiation)

4.3. Observation Results through Data Visualisation

During the observation in the three-dimensional data visualisation space, we noticed several compact clusters with exceptional colour properties, such as being much brighter and more colourful than others. Among the one-hundred clusters labelled from 1 to 100, we selected ten outstanding ones whose identification numbers were *3, 25, 26, 85, 11, 14, 80, 53* and *86*, respectively, and the top-three frequently used hashtags of each of these ten clusters were listed in **Table 3**. The order of items in the table had been adjusted to show that cluster *3, 25, 26, 85* were attached with all-positive hashtags; cluster *11, 14, 80* were attached with all-negative hashtags, and that cluster *43, 53, 86* were attached with hashtags of either positive or negative sentiment.

Table 3: Representative hashtags of each observed cluster

	Top 3 Representative Hashtags		
c 3	delicious	relax	happiness
c 25	smile	beautiful	love
c 26	love	beautiful	smile
c 85	relax	love	beautiful
c 11	cry	depress	upset
c 14	sad	upset	heartbroken
c 80	stress	sad	hate
c 43	cry	smile	beautiful
c 53	hate	love	inspiration
c 86	heartbroken	tear	love

The Hue analysis results of the ten observed clusters were plotted in a combined line chart in **Figure 19**. The overall hue pattern of the observed clusters was similar to that of groups generated by Hierarchical clustering, generally with the first peak at Orange and second peak at Azure. However, the scale range of the proportion was much bigger, which showed larger difference of hue between clusters; moreover, relatively more distinctive hue patterns could be found in the observed results. Cluster *3* (*c3*) showed the highest proportion in Orange at more than 50% while had the lowest proportion in Red and Azure; cluster *43*, however, had the best performance in Red while showed the least proportion in Orange among all clusters; cluster *85* stood out in Azure at around 17% and cluster *86* performed prominently in Blue at about 11%.

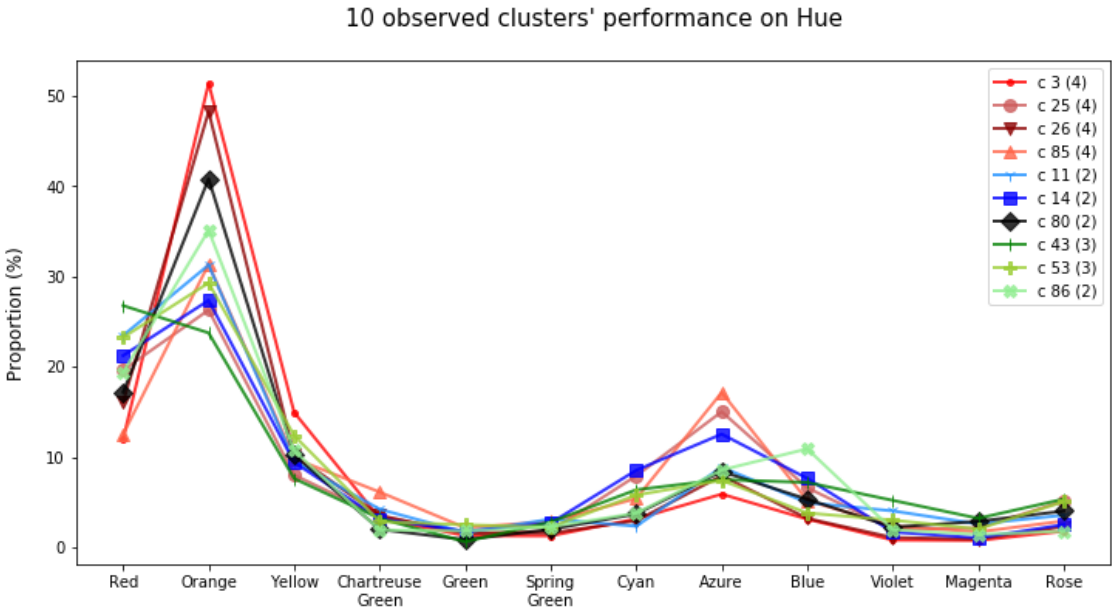


Figure 19: Ten observed clusters' performance on Hue.
The number in the parentheses shows the resulting group of Hierarchical clustering each observed cluster belongs to. (The colour of lines is only for differentiation)

The Saturation and Value analysis results of the ten observed clusters were illustrated in a combined scatter plot in **Figure 20**, and the mean value and ninety-five percent confidence interval were elaborated in **Table 4**. As shown in the figure and table, cluster *3* and *26* had significantly higher Saturation and Value compared with most of the other clusters; cluster *43* showed the lowest mean value in Saturation while had relatively high brightness (Value); although the range of 95% confidence interval of cluster *86* was the largest, it still had a much lower average Value among all.

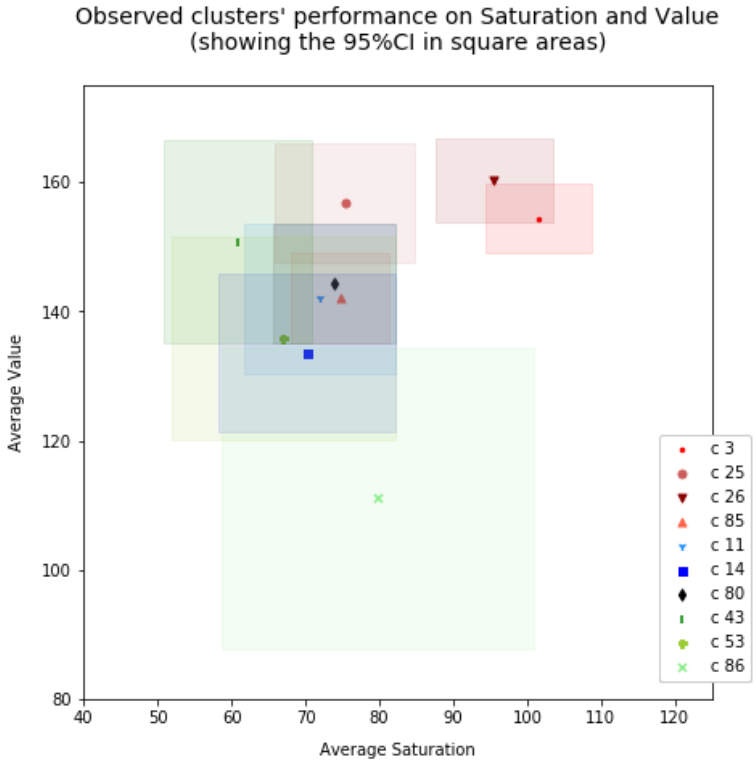


Figure 20: Ten observed clusters' performance on Saturation and Value
The square area shows the co-region of 95% confidence interval of mean Saturation and mean Value. (The colour of points and squares is only for differentiation)

Table 4: Saturation and Value statistics of ten observed clusters

	Saturation		Value	
	Mean	95% CI	Mean	95% CI
c 3	101.53	[94.25, 108.81]	154.37	[148.98, 159.77]
c 25	75.33	[65.93, 84.73]	156.86	[147.60, 166.11]
c 26	95.53	[87.66, 103.40]	160.26	[153.76, 166.75]
c 85	74.68	[67.99, 81.38]	142.06	[135.02, 166.75]
c 11	71.94	[61.71, 82.18]	142.04	[130.44, 153.64]
c 14	70.24	[58.25, 82.24]	133.61	[121.39, 145.83]
c 80	73.90	[65.61, 82.20]	144.36	[135.19, 153.53]
c 43	60.84	[50.88, 70.80]	150.94	[135.18, 166.69]
c 53	67.02	[51.87, 82.17]	135.88	[120.11, 151.64]
c 86	79.70	[58.59, 100.82]	111.05	[87.64, 134.46]

Section 5: Discussion, Limitations and Outlooks

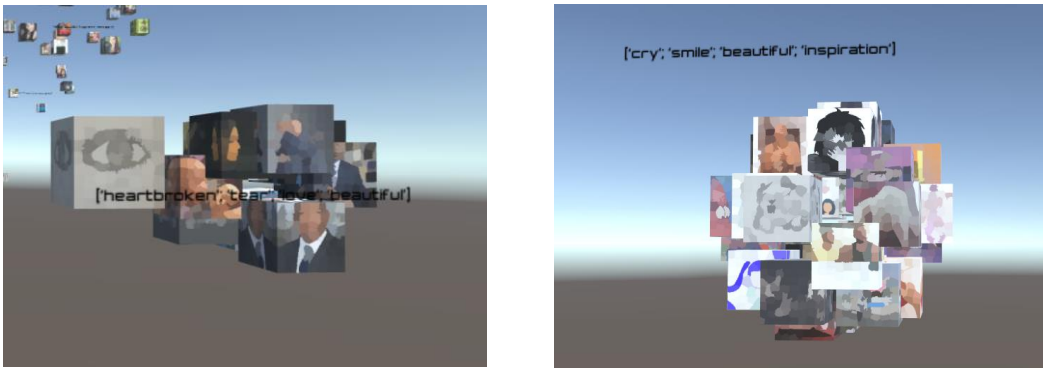
The colour analysis result shows a correlation between the mean value of saturation/*value* and groups of different semantic properties, while it is challenging to find such association for hue due to the difficulty of defining the "averaged" hue. For one thing, when calculating the averaged hue by Means of Circular Quantities method, the sample means are normally distributed due to the Central Limit Theorem, which leads to approximately the same means of all groups that lie in the hue interval of Red. For another, the perception of hue can be affected by various factors, such as the background color [32], and hence the hue of an image perceived by human perceptual system might not be the same as the computational averaged one, which draws the issue that whether we should interpret the averaged hue based on algorithm-computed result or human perceptual result. An experiment recorded in a previous study allows the participants to adjust colour stimulus until they think the perceived hue equals the specified reference hue [33]; it turns out that the results from different observers are not the same. Although the difference is slight, it shows that the perception of hue varies with the personal psychophysical qualities (the wavelength of different hue also matters). Therefore, the complexity of colour cross-correlation is worthwhile to be taken back for further investigation into the average hue interpretation.

The performance of different hue tells another story. Overall, the hue distribution of images follows a general pattern with the most appeared hue being Orange and Red and the second most used hue being Azure. Along with the general pattern, we can still find unique features in certain groups; however, groups from Hierarchical clustering may contain several close-packed groupings inside. Owing

to the three-dimensional data visualisation tools, many small clusters with exceptional colour patterns can be noticed. For example, during the analysis of observed clusters, cluster *3* (**Figure 21 (a)**) had a major hue of Orange and cluster *85* (**Figure 21 (b)**) performed prominently in Azure, yet both of them belong to *Group4*, originally. The colour analysis of the Hierarchically clustered groups considered data points scattered far away from the core groupings and generalised a group's hue performance, whereas a proper visualisation technique like what we used can help us identify more compact clusters and avoid the analysis sensitivity to outliers. Meanwhile, we can explore more specific details about clusters' semantic relationship with their colour patterns, such as that the most representative hashtag of cluster *3* is "delicious" and for cluster *85* it is "relax". The other two conspicuous clusters observed are cluster *86* (**Figure 21(c)**) and cluster *43* (**Figure 21(d)**); the former shows an average low *value* while the latter shows an average low saturation but high *value*. With the hashtag labels above each observed cluster, observer can conclude the association between the visual and textual contents and bring up further questions on them more intuitively.

(a)

(b)



(c) (d)

Figure 21: The observed clusters in the Unity
Note that (a) is cluster 3, (b) is cluster 85, (c) is cluster 86 and (d) is cluster 43. The labels in brackets show the representative hashtags of each clusters.

Earlier research had investigated the link between colour and psychological functioning. Since most of the extant research conducted in pre-twenty-first century failed to systematically look into the multidimensionality of colour stimuli [34], the psychological influence of colour was concluded in a more general way. A speculation indicates that plus colours, such as red and yellow, tend to induce positive feelings and prompt an outward focus, whereas minus colours, such as blue, can induce negative feelings and encourage an inward focus [35, 36]. However, more recent studies point out that it is difficult to establish firm rules about emotional response based on single, or even two colours [37-39]. All colours are open to a myriad of interpretations, depending on the context and the way they are combined; thus, there are emerging colour-in-context theories in the past decade linking the colour usage with specific backgrounds, such as analysis from biological perspectives or during certain experience, to elaborate on the colour psychological phenomenon [34].

Our research attends to the association of colours and the specified semantic

meanings while has not progressed on the exploration of the psychological effect. The data collected during our research are all from the owners who were willing to share their personal life to the public. For further investigation in the colour psychological field, it may be worthwhile to probe into the characteristics of the data owners and combine the findings with the colour semantic relationship to make more profound discoveries.

The visualisation method used in this research assists in the colour pattern discovery under different categorisation algorithms. It would be of interest to apply the similar visualisation way in diversified fields. For example, based on the existing theory of physical property of chemicals, a pharmaceutical industry may be able to use the visualisation to detect abnormal performance of certain drug material under a precisely defined chemical classification procedure. For more general industries, one can use this kind of visualisation to find out a pattern showing how the different business or marketing performance links to colour usages in the companies' official platforms or relevant advertisements, and gain clues for generating better design.

Section 6: Conclusions

In this project, we explore the association between the colour pattern and the attached hashtags of images from the social media platform, Instagram. The hashtags are served as the sentiment label of images to see if the users tend to express emotions through images of certain colour properties.

To accomplish this, a feature vector is generated for each image data point based on the selected sixteen hashtags, and an agglomerative Hierarchical clustering method is then applied to these features to classify images with similar hashtag properties together. we conduct a colour analysis on the images of resulting five groups in respect of their hue, saturation and *value* by extracting and averaging the HSV values of pixels. The results show that the group of images attached with only positive hashtags, such as *love*, *beautiful*, *smile*, *amaze*, *inspiration*, *delicious*, *happiness* and *relax*, tend to include more orangish colour and have significantly higher saturation and *value* than images in other groups. Images with *sad* accounting for a much larger proportion among the attached hashtags tend to include more blue tone and have the lowest *value* among all sample images.

On the other hand, a three-dimensional data visualisation is implemented for gaining further insights into the colour patterns of the sample image data. During the navigation through the 3D virtual reality space, several smaller clusters of images with exceptional colour properties are noticed by human eyes, and it is found that some clusters with perceivable colour difference are classified into the same group by the previous clustering algorithm. The visualisation method like this can help detect more specific groupings and prevent the colour and semantic

analysis from being affected by the abnormal properties of outliers. With appropriate information, such as the representative hashtags, labelled above each observed image clusters, one can generate a more intuitive understanding of the relationship between observed elements in a more efficient way.

References

1. Orben, A.C. and R.I. Dunbar, *Social media and relationship development: The effect of valence and intimacy of posts.* Computers in Human Behavior, 2017. **73**: p. 489-498.

2. Kessler, S., in *The photo economy*. 2014, Fast Company. p. 54–60.

3. Liu, B., *Sentiment analysis and opinion mining.* Synthesis lectures on human language technologies, 2012. **5**(1): p. 1-167.

4. Over, P., et al., *TRECVID 2012-an overview of the goals, tasks, data, evaluation mechanisms and metrics.* 2013.

5. Borth, D., et al. *Large-scale visual sentiment ontology and detectors using adjective noun pairs*. in *Proceedings of the 21st ACM international conference on Multimedia*. 2013.

6. Ferrara, E., R. Interdonato, and A. Tagarelli. *Online popularity and topical interests through the lens of instagram*. in *Proceedings of the 25th ACM conference on Hypertext and social media*. 2014.

7. Vellido Alcacena, A., et al. *Seeing is believing: The importance of visualization in real-world machine learning applications*. in *Proceedings: 19th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2011: Bruges, Belgium, April 27-28-29, 2011*. 2011.

8. Wang, X., et al., *Modeling emotion influence in image social networks.* IEEE Transactions on Affective Computing, 2015. **6**(3): p. 286-297.

9. Solli, M. and R. Lenz. *Color emotions for image classification and retrieval*. in *Conference on Colour in Graphics, Imaging, and Vision*. 2008. Society for Imaging Science and Technology.

10. Amencherla, M. and L.R. Varshney. *Color-based visual sentiment for social communication*. in *2017 15th Canadian Workshop on Information Theory (CWIT)*. 2017. IEEE.

11. Valdez, P. and A. Mehrabian, *Effects of color on emotions.* Journal of experimental psychology: General, 1994. **123**(4): p. 394.

12. Smilkov, D., et al., *Embedding projector: Interactive visualization and interpretation of embeddings.* arXiv preprint arXiv:1611.05469, 2016.

13. Shmueli, G., et al., *Data mining for business analytics: concepts, techniques, and applications in R*. 2017: John Wiley & Sons.

14. Gao, X.P., et al., *Analysis of cross-cultural color emotion.* Color Research & Application: Endorsed by Inter-Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur, 2007. **32**(3): p. 223-229.

15. Thelwall, M., *Heart and soul: Sentiment strength detection in the social web with sentistrength, 2017.* Cyberemotions: Collective emotions in cyberspace, 2014.

16. Stilo, G. and P. Velardi, *Hashtag sense clustering based on temporal similarity.* Computational Linguistics, 2017. **43**(1): p. 181-200.

17. Peikari, M., et al., *A cluster-then-label semi-supervised learning approach for pathology image classification.* Scientific reports, 2018. **8**(1): p. 1-13.

18. Bhagat, A., et al., *Penalty parameter selection for hierarchical data stream clustering.* Procedia Computer Science, 2016. **79**: p. 24-31.

19. Kuhlmann, M., et al., *Mixing positive and negative valence: affective-semantic integration of bivalent words.* Scientific Reports, 2016. **6**(1): p. 1-7.

20. Theus, M., *High-dimensional data visualization*, in *Handbook of data visualization*. 2008, Springer. p. 151-178.

21. Lee, Y.K., E.R. Lee, and B.U. Park, *Principal component analysis in very high-dimensional spaces.* Statistica Sinica, 2012: p. 933-956.

22. Maaten, L.v.d. and G. Hinton, *Visualizing data using t-SNE.* Journal of machine learning research, 2008. **9**(Nov): p. 2579-2605.

23. Rousseeuw, P.J., *Silhouettes: a graphical aid to the interpretation and validation of cluster analysis.* Journal of computational and applied mathematics, 1987. **20**: p. 53-65.

24. Ibraheem, N.A., et al., *Understanding color models: a review.* ARPN

Journal of science and technology, 2012. **2**(3): p. 265-275.

25. Fairchild, M.D., *Color appearance models*. 2013: John Wiley & Sons.

26. Poorani, M., T. Prathiba, and G. Ravindran, *Integrated feature extraction for image retrieval.* International Journal of Computer Science and Mobile Computing, 2013. **2**(2): p. 28-35.

27. Ford, A. and A. Roberts, *Colour space conversions.* Westminster University, London, 1998. **1998**: p. 1-31.

28. Jammalamadaka, S.R. and A. Sengupta, *Topics in circular statistics*. Vol. 5. 2001: world scientific.

29. Bárány, I. and V. Vu, *Central limit theorems for Gaussian polytopes.* The Annals of Probability, 2007. **35**(4): p. 1593-1621.

30. Brown, A.M., D.T. Lindsey, and K.M. Guckes, *Color names, color categories, and color-cued visual search: Sometimes, color perception is not categorical.* Journal of vision, 2011. **11**(12): p. 2-2.

31. Webster, M.A. and P. Kay, *Color categories and color appearance.* Cognition, 2012. **122**(3): p. 375-392.

32. Rosenholtz, R., A.L. Nagy, and N.R. Bell, *The effect of background color on asymmetries in color search.* Journal of vision, 2004. **4**(3): p. 9-9.

33. Webster, J., P. Kay, and M.A. Webster, *Perceiving the average hue of color arrays.* JOSA A, 2014. **31**(4): p. A283-A292.

34. Elliot, A.J. and M.A. Maier, *Color psychology: Effects of perceiving color on psychological functioning in humans.* Annual review of psychology, 2014. **65**: p. 95-120.

35. Von Goethe, J.W., *Theory of colours*. Vol. 3. 1840: Mit Press.

36. Goldstein, K., *Some experimental observations concerning the influence of colors on the function of the organism.* Occupational Therapy, 1942.

37. Ou, L.C., et al., *A study of colour emotion and colour preference. Part I: Colour emotions for single colours.* Color Research & Application, 2004. **29**(3): p. 232-240.

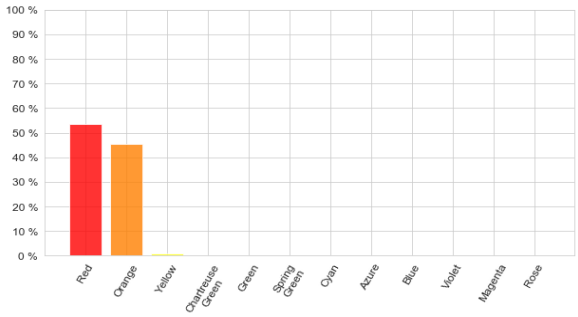
38. Ou, L.C., et al., *A study of colour emotion and colour preference. part II: colour emotions for two-colour combinations.* Color Research &

Application: Endorsed by Inter-Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur, 2004. **29**(4): p. 292-298.

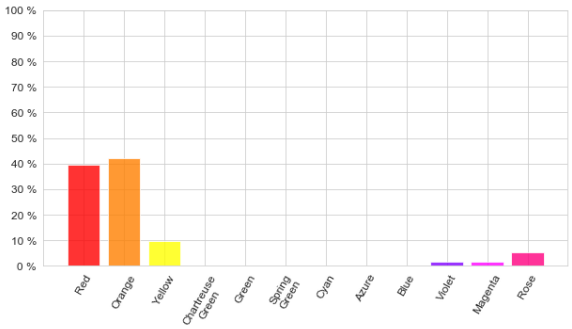
39. Ou, L.C., et al., *A study of colour emotion and colour preference. Part III: Colour preference modeling.* Color Research & Application, 2004. **29**(5): p. 381-389.

Appendix

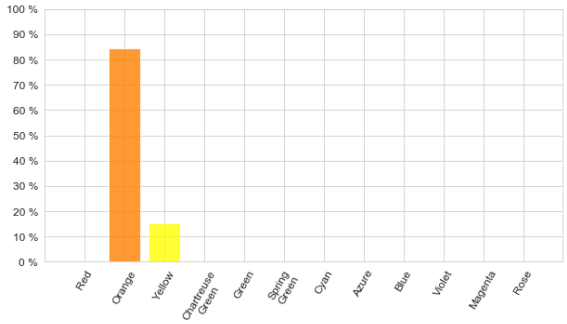
1. Sample hue detection results by OpenCV module



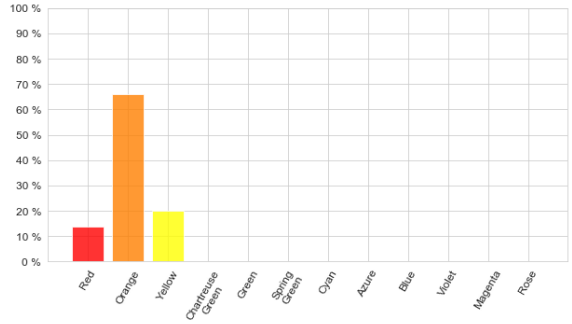
(a)



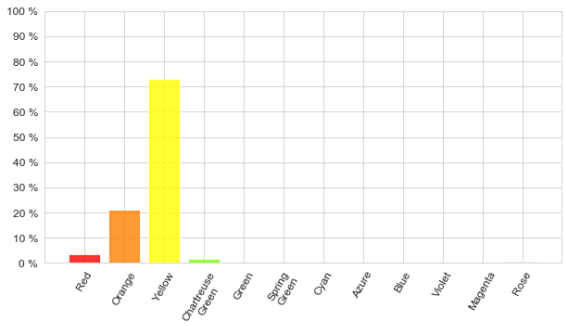
(b)



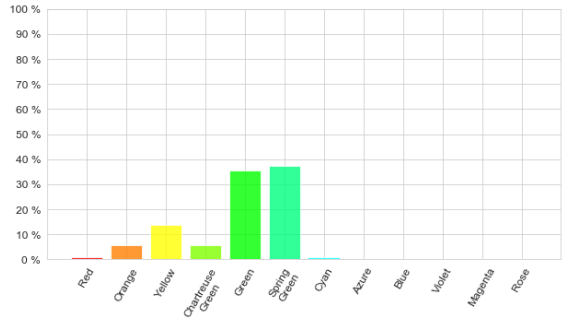
(c)



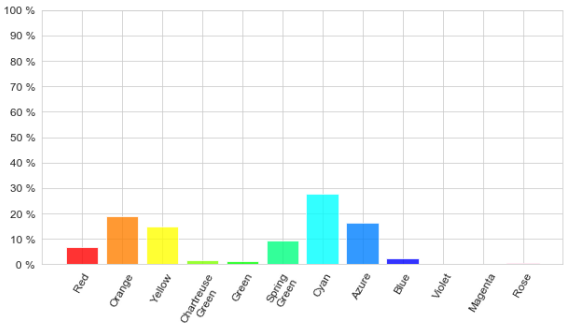
(d)


100 %
90 %
80 %
70 %
60 %
50 %
40 %
30 %
20 %
10 %
0 %
Red
Orange
Yellow
Chartreuse Green
Green
Spring Green
Cyan
Azure
Blue
Violet
Magenta
Rose

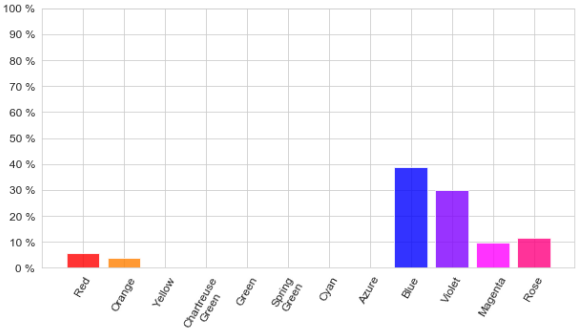
(e)


100 %
90 %
80 %
70 %
60 %
50 %
40 %
30 %
20 %
10 %
0 %
Red
Orange
Yellow
Chartreuse Green
Green
Spring Green
Cyan
Azure
Blue
Violet
Magenta
Rose

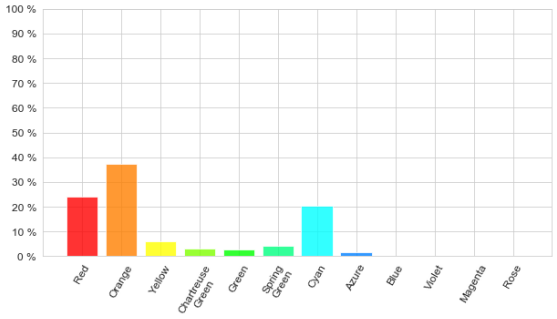
(f)


100 %
90 %
80 %
70 %
60 %
50 %
40 %
30 %
20 %
10 %
0 %
Red
Orange
Yellow
Chartreuse Green
Green
Spring Green
Cyan
Azure
Blue
Violet
Magenta
Rose

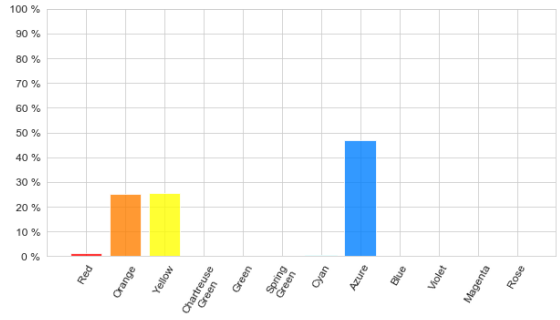
(g)


100 %
90 %
80 %
70 %
60 %
50 %
40 %
30 %
20 %
10 %
0 %
Red
Orange
Yellow
Chartreuse Green
Green
Spring Green
Cyan
Azure
Blue
Violet
Magenta
Rose

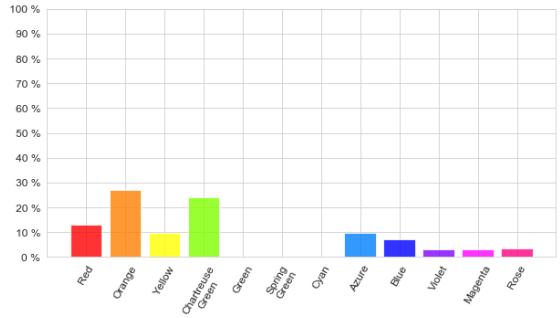
(h)



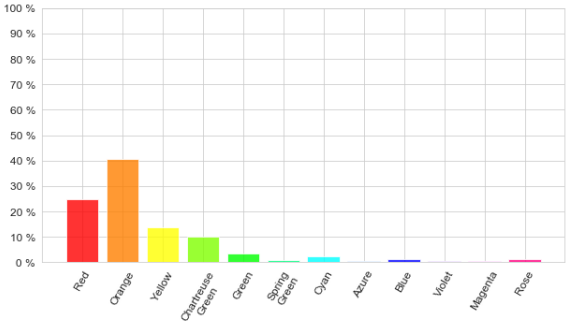
(i)



(j)



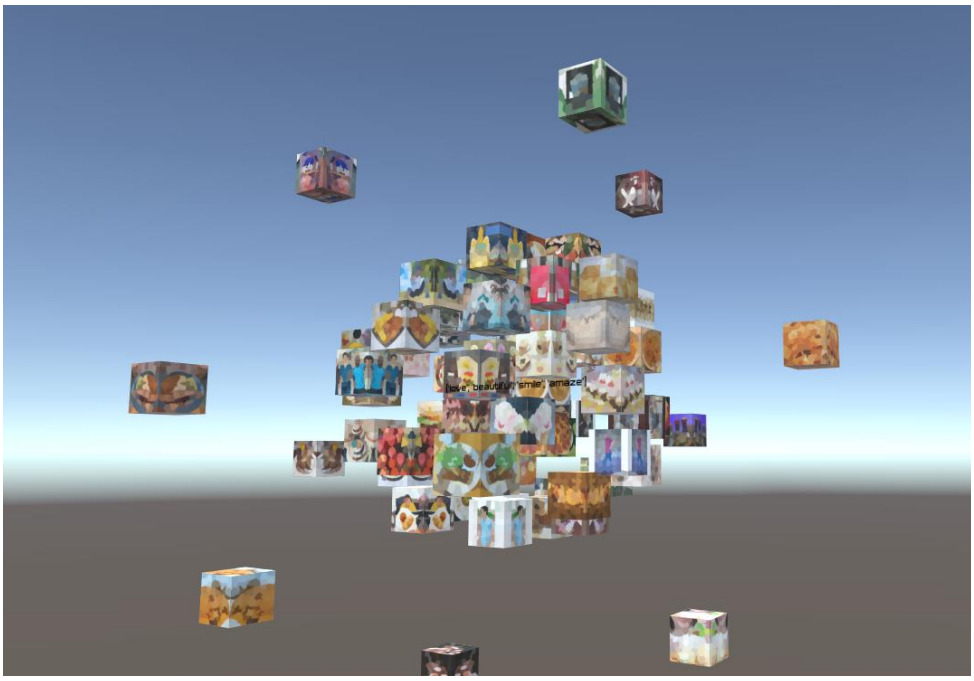
(k)



(l)

Figure 22(a) – (l): Sample hue detection results by OpenCV module

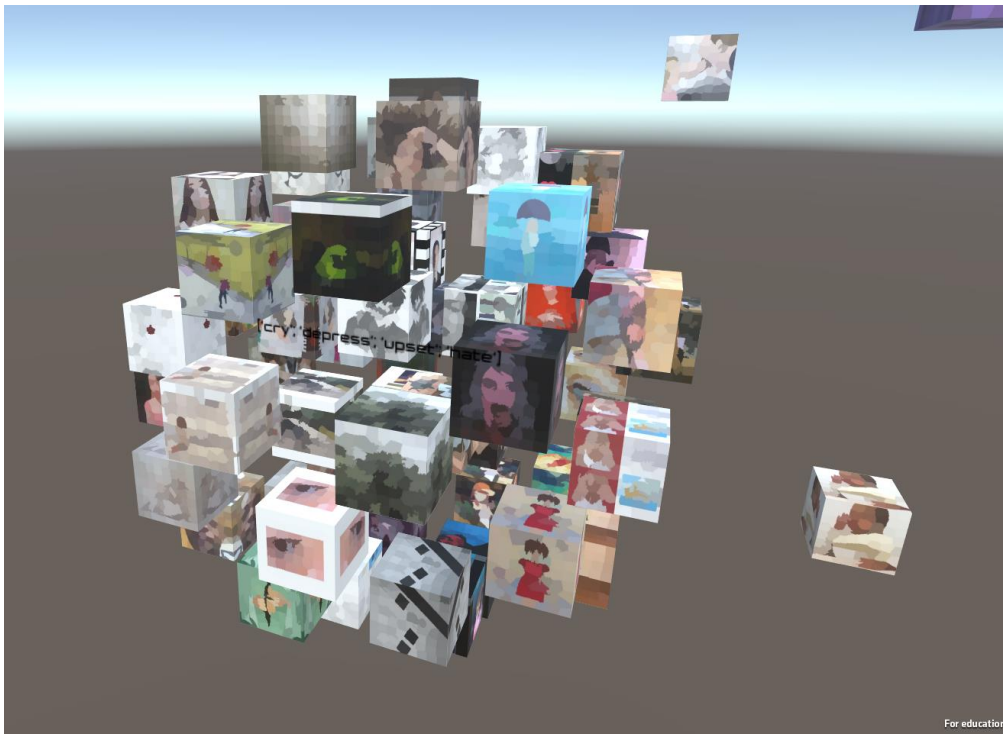
2. Other observed clusters in the 3D visualisation space



(a) Representative hashtags: *love, beautiful, smile, amaze*

(b) Representative hashtags: *smile, beautiful, love, happiness*

(c) Representative hashtags: *beautiful, delicious, love, smile*



(d) Representative hashtags: *cry, depress, upset, hate*

(e) Representative hashtags: *sad, upset, heartbroken, hate*

(f) Representative hashtags: *hate, love, inspiration, tear*

Figure 23(a) – (f): Other observed clusters in the 3D visualisation space