**Progress Report**

**Speech Bubble Detection**

**<Junaid Suleman, Wenbo Wu, Pengxiang Jia>**

**<March 15th, 2021>**

**Contents**

[**Introduction**](#_vou0ffdbqpcp) **3**

[1.1 Purpose](#_hblqvwmbyruc) 3

[1.2 Project Scope](#_qhwe6e2mes3o) 3

[1.3 Datasets](#_jctvrcygmepc) 3

[1.3.1 Manually Create Datasets](#_b02cf1tk3e0v) 3

[1.3.2 Expert System](#_pwfap1kyxp0m) 3

[1.3.3 Semi-supervised Learning](#_k1o17e8d53z8) 4

[**Overall Description**](#_36s6vxhgqddf) **4**

[2.1 Goals](#_drp5xlxo8uy5) 4

[2.2 Plan](#_poszv079mshd) 4

[2.3 Progress to Date](#_2bk8tjay6do4) 5

[**2.4 Task Breakdown**](#_exyfi68hm2rr) **6**

[**3. Initial Results**](#_xne971h2xlmh) **7**

[**Reference**](#_bxp5jor9bfvf) **11**

## **Introduction**

### **1.1 Purpose**

There are many ways to kill time in today’s society. Many modern people like to kill time by reading comics and watching movies including anime and TV shows, especially during a time period where people are forced to stay home and find various hobbies to spend their time and as such the consumption of comic books, mangas, manhwa, etc. has increased. And a lot of the time these books are written in languages other than the reader's native language.

Therefore, translating these books from one language to another is a lot of work. As such with this project we wish to attempt to streamline the process of possibly fetching the text from these books into an easy-to-read format that the translators can access without having to browse through all the pages. In addition, the detection part of the process can also be used to automatically remove text from images streamlining the process even further.

Moreover, with the development of the comic eras, the demand for manga adapted into anime is increasing. Because there is a big gap in speed between production anime and drawing manga. We wish that this project will have a corresponding effect on the development of comics in future. Also researchers believe that speech balloon detection systems have potential for applications such as preparing digitized comics for computerized presentation on handheld devices, or developing assistive technologies for people with low vision.

### **1.2 Project Scope**

The dataset for bubble detection is the more complicated of the two, and as such we haven’t been able to acquire a dataset that we find is appropriate for our use case, however we plan to dig around and hope to find a dataset that is appropriate for our use case. In the event where we are unable to acquire such a dataset we can either choose to create it ourselves or we can go the route of image processing with an expert system to automatically create the training data and labels required for the selected method of machine learning.

### **1.3 Datasets**

So far finding datasets hasn’t been successful and as such we might just have to create the dataset by hand which will take quite a bit of time. We also thought using an expert system and semi-supervised learning.

#### **1.3.1 Manually Create Datasets**

This will be a relatively large amount of work and the most primitive method. We will collect all kinds of comics images including grayscale images, color scale images, etc and then manually highlight the speech bubble for training purposes.

#### **1.3.2 Expert System**

During research, we found related papers on image processing. With the aid of the Expert System, we can implement a program that divides the image into multiple regions and then extract the speech bubble from each region. So as to use this system to automatically generate a dataset for our use. We’ve also taken the approach of simplifying the Expert system to just be hurestical-features so that we don’t have to implement a fully fledged expert system and instead may be able to set up feature identification based on set values.

#### **1.3.3 Semi-supervised Learning**

Semi-supervised Learning is the hybrid of learning using both labelled and unlabelled data, it is also a popular method, so we consider using it because we lack labelled data and manually labeling all data is not the optimal choice.

## **Overall Description**

### **2.1 Goals**

The goal of this project is to produce a model or models that can successfully detect speech bubbles and highlight the bubble area.

2.1.1 Measure of Success  
  
Level 1: The system is able to detect something in the image. The behavior of the system is constant but the results are unpredictable. E.g. The system can indicate an element, but it is not always a speech bubble.

Level 2: The system is able to detect speech bubbles in some or most of the cases, but it makes mistakes sometimes.

Level 3: The accuracy rate is as high as 90% on detection and it can highlight the bubble area.

Level 4: The system can persist having a high accuracy rate on new inputs of manga/comic pages.

2.1.2 Discussion of Measure of Success

One of the main indicators of success for this project is the detection of a speech bubble given the input image. The program should be able to find and classify every single speech bubble in the provided image. Formally speaking, it should be able to correctly detect all speech bubbles and render them out separately with its content intact, regardless of the image color.

The next measure of success is the accuracy of which it can present the bubble. It is important that the error in this step is as miniscule as possible, since if it isn’t properly able to detect the edge then it might miss or exclude some of the text in the bubble, decreasing the overall accuracy of the program.

The model delivers state-of-the-art performance with an F1-score of over 0.94. Once the overall accuracy is maximized, we can begin to reduce the time and computing power required to recognize bubbles and characters and construct an overall accuracy score and success score for the project/program.

We have a chance to scale back the speech bubble and speech text detection back to only speech bubble detection because doing the both may be too many goals, so we need to focus on the primary one.

### **2.2 Plan**

For this project, we plan to first obtain reliable test data that can be used to create the program itself. This includes pdf and/or images that can be used as input to create the basic code of the program. Once the test data is acquired, we can use it to determine how to import the image into the program in a manner that we can then manipulate and extract features from. After accessing the binary data of the image we would then need to clean the image and make sure there is as little noise as possible followed by converting the image to only black or white pixels instead of a range of pixels.

Then, we select a region that might count as a speech bubble and extract features from the region which are then fed into the machine learning algorithm.

We plan to experiment with multiple different algorithms that we learned in the course to help us identify the bubbles and select the most suitable algorithm based on F-score generated by each algorithm. After we’re satisfied with the detection part of the program, we can start language recognition. For language recognition, we hope to be able to extract features from the selected region based on the dataset [2]. That particular dataset will be used to train our machine learning algorithms. We can then carry out experiments pertaining to related F-scores of each algorithm and the time and space required to correctly classify each subsequent test data, choosing the most optimal one for the final implementation.

### **2.3 Progress to Date**

So far we’ve gone through the process of getting an image into the program, convert it into a numpy array and perform manipulations on it.

A grayscale filter is applied to the image and then binary threshold is applied to clean up the image. The resulting threshold is then processed through the Run Length Smoothing Algorithm that bunches up the various white and black pixels together removing shades of grey, leaving clusters of black and white pixels for easier extraction of elements of interest. Then using sklearn-image we apply segmentation via measure (a skimage package) to label to extract the bubbles and thus creating semi-prepared labels that are currently colored in and need to be converted to black and white.

Next we’ve managed to extract some semblance of features from the rlsa image using the ORB algorithm (an alternative for SIFT algorithm, which is privately owned).

## 

## **2.4 Task Breakdown**

The task breakdown can be seen below in the order they appeared in the plan above with machine learning tasks in bold. As one can see all group members are responsible for at least one machine learning task.

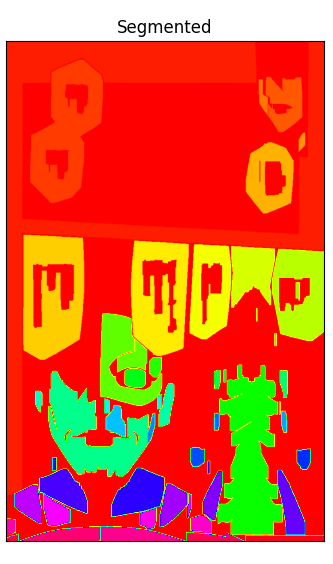
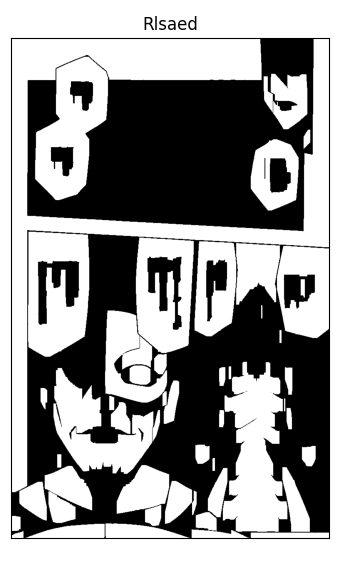
## 

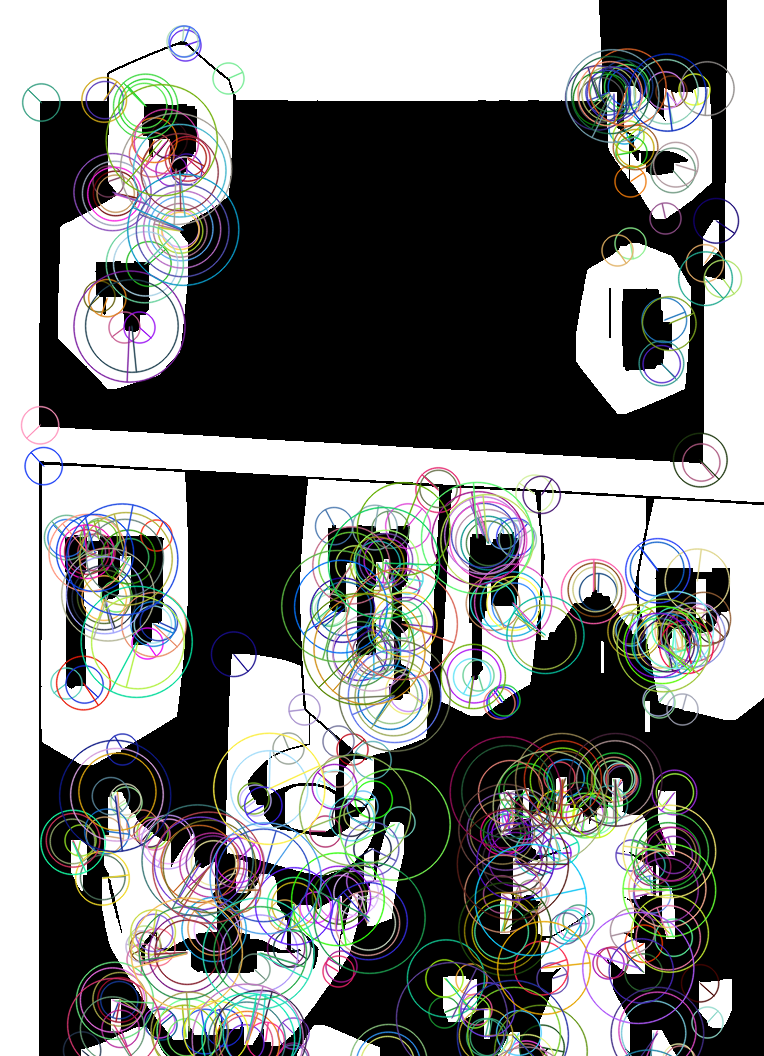
| ID | Task Name | Start Date | End Date | Duration | Percent Complete | Members (alphabetical order) |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | Data Selection and Data Cleaning | 2021-02-16 | 2021-03-01 | 15 Days | 15% | None |
| 2 | Image Segmentation | 2021-03-01 | 2021-03-13 | 12 Days | 80% | Junaid |
| 3 | Feature Extraction | 2021-03-15 | 20 21-03-19 | 4 Days |  | Pengxiang, Wenbo |
| 4 | Use Features to Build ML Model for Speech Bubble Detection | 2021-03-20 | 2021-03-25 | 5 Days |  | Junaid, Pengxiang, Wenbo |
| 5 | Model Selection and Hyperparameter tuning | 2021-03-25 | 2021-03-30 | 5 Days |  | Junaid, Wenbo |
| 6 | Verify and Visualization | 2021-04-01 | 2021-04-05 | 5 Days |  | Pengxiang, Wenbo |
| 7 | Project Presentation | | 2021-04-06, 07, 08 | | | |
| 8 | Final Report | 2021-04-09 | 2021-04-12 | 3 days |  | Junaid, Pengxiang, Wenbo |

## 

## **3. Initial Results**







Feature extraction drawn as key points and descriptors via ORB

## 

## **Reference**

[1] Kuboi, Toshihiro. (2014). ELEMENT DETECTION IN JAPANESE COMIC BOOK PANELS. 10.13140/2.1.1009.4409.

[2] Dubray, David & Laubrock, Jochen. (2019). Deep CNN-based Speech Balloon Detection and Segmentation for Comic Books. Retrieved from: [1902.08137v1.pdf (arxiv.org)](https://arxiv.org/pdf/1902.08137v1.pdf)