The Recognition of Characters and Speech Balloons in Japanese Comics Based on Deep Learning

Tengfei, Shao
Tianjin Polytechnic University

I. Research Background

In recent years, instead of paper comics, more and more people have started to use electronic comic (e-books). According to the research[1], The e-book market in 2015 is estimated at 158.4 billion yen, which increased 31.8 billion yen (25.1%) from 126.6 billion yen. Including e-books and e-magazines, the electronic publishing market reached 182.6 billion yen and is still expanding. E-comics contributed to the increase of more than 30%[2] in e-book market in 2015, while the paper comics showed a decline of nearly 7%. In 2016, e-comic market is also in a high growth rate and rose 27.1%[3] from 2015. However, paper comic market is still in a downward trend and declined 7.4% from 2015. The e-comics dominates 82%[4] market share of e-book market. Therefore e-comic is a very important industry and has great potential for development. As the scale of e-comics market increases, it is significant to improve the service and the convenience of reading supporting system for e-comics.

Now the reading supporting system for e-comics is not enough robust. For example, the comic search systems of most websites can only be searched by the name of the comic. In addition, the e-comics on e-comic website, are mostly collected by scanning from paper comics, which means people cannot directly extract texts of speech balloons, characters models and frames in comic pages. People have to extract the components of manga manually and it will cost a lot of time(inefficiency). In order to reduce the cost of extracting components from metadata and build a base for new services of e-comics, such as searching by images and automatically changing the size of speech balloons(like Bubble Zoom of Google Play app), it is necessary to propose a new model that can automatically the components of manga.

The extractions of manga mainly contains speech balloon, face, character and frame. Tanaka et al.[5] used Ada-Boost to identify texts and detected speech balloons by finding white areas of speech balloons. Arai K et al.[6] used modified connected component labeling(CCL) method to detect frames and speech balloons. Yusuke et al.[7] proposed a method that can detect face of characters by edge orientation histogram(EOH). Sun, W et al.[8] recognize Cartoon character by Histograms of Oriented Gradients (HOG). Chu, W. T. et al.[9] proposed a Manga FaceNet(based on deep neural network) to finish the detection of character's face. Thanks to these papers, we can get a conclusion that now the method based on deep learning can get a good performance(most in accuracy) in extraction of manga components. Therefore, we plan to propose a model based on deep learning, which can automatically extract both speech balloons and characters.

II. Research Purpose

For most of the e-comics are scanned from paper comics, people cannot extract the components of e-comics efficiently and accurately. Moreover, different from the color information of natural images is very rich, the information we can use in most manga is black-and-white and some gray pixels. So not

many researchers want to do researches about manga. But if we want to implement more practical functions, such as images search system and Bubble Zoom function(**Fig.1**) for e-comics, the basic models and algorithms are necessary. Therefore, in order to reduce the cost of extractions and promote the development of e-comics, we plan to propose a model based on Faster r-cnn algorithm to challenge this problem. The model can extract character and speech balloon automatically.

@Stan Lee

Fig.1 Bubble Zoom function of Google Play app. The size of speech balloons can changes automatically. Images are from Google Play.

III. Prior Research

A. Convolutional neural network

Convolutional neural networks (CNN) is a deep neural network with a convolutional structure. The nature of CNN is by building deeper and deeper hidden layers, which can adaptively[10] obtain features of images. Thorough training the CNN by a lot amount of labeled data, CNN can extract low-dimensional semantics of images, which are highly abstracted features that humans cannot recognize. So compared with recognizing the images by the artificial features, trained by a huge amount of data CNN is more powerful in depicting the rich internal information of images and can recognize them in a high accuracy. Thus, We plan to use the CNN as a basic network in the extraction model. As **Fig.2** follows. The classic CNN structure is LeNet-5.

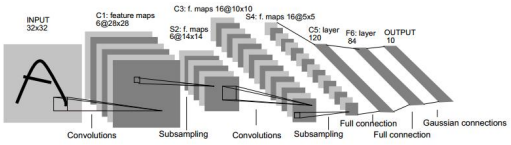


Fig2. LeNet-5 Model. From paper[11].

B. Faster R-CNN

The region-based neural network has developed rapidly recent years, especially in the filed of object detection and semantic segmentation. After R-cnn [12] and Fast R-cnn [13], in order to reduce the time of generating the proposal regions, Microsoft Girshick et al.[14] proposed the a new object detection algorithm, Faster r-cnn. We can see Faster r-cnn as a Fast R-cnn model with a PRN net. The RPN net replaced the methods, such as Selective Search[15] and EdgeBoxes[16], for RPN can use NN to generate proposal regions more efficiently. Faster r-cnn has achieved a significant accomplishment in speeding up the generating of proposal boxes and its accuracy has reached the most advanced level in 2015. The Faster r-cnn algorithm has enabled that the four basic steps of object detection (candidate region generation, feature extraction, classification, location refinement) can be unified within a deep network framework and all of them can be computed on GPU. Faster r-cnn based model achieved the best accuracy of object recognition in PASCAL VOC2007 and PASCAL VOC2012 at that time. Also it is the winner of MS COCO(Microsoft Common Objects in Context) in 2015 year.

On the base of Faster r-cnn algorithm, we plan to proposal an extraction model. The **Fig.3** is an overview of extraction model. Different from usual Faster r-cnn, we replace the VGG network by Resnet network, which can get a better performance[17] in object detection. Simultaneously, it requires more powerful graphics cards. Firstly, we do some preprocessing of images and send them into

Resnet. Through doing convolution of the image, Resnet can get low-dimensional vector features and learn their semantics. Then, depending on the abstract of the image, Resnet can predict the class and position of the object by using softmax classification and Bbox regression functions.

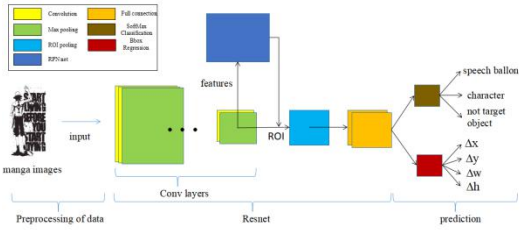


Fig.3 Structure of extraction model.

C. Region Proposal Networks

1) Sliding window

RPN (Region Proposal Network) can generate candidate regions by neural network more efficiently[18] than other methods do, such as Selective Search and Edge Boxes. As **Fig.4** shows, RPN uses a "sliding window" structure and adds it to the last shared convolution layer(Feature Map:FM) of the CNN . The sliding window is a 3 * 3 size classifier and connected with the FM in the form of a full connection. Sliding window outputs low-dimensional vectors mapped from CNN and send them to two networks (reg-layer and cls-layer), which are fully connected with sliding window. The output consists of two parts[8], "objectness score" (2 k score) and "object proposals"(4 k coordinates) . Thereafter, the two layers(reg-layer and cls-layer) will process the data of object proposals and objectness scores. The Reg layer predicts the coordinates of "central anchor of proposals" and its real object position (coordinates x, y, and height w, h). The Cls layer determines whether the generated region is an object or a background. When the 3 * 3 size sliding window slides on the FM, each slid position corresponds to k different anchor boxes of the original image. Therefore, the Reg layer outputs a vector of 2 * k dimension (correspond to the score of the target and the background), and the Cls layer outputs a vector of 4 * k dimension (correspond to the conversion parameter of the anchor box). The sliding window guarantees the correlation between two layers(reg, cls) and image features. Since RPN and Fast-RCNN can share the features in the same CNN, the time of generating proposal regions has been greatly reduced. In this plan, we will use RPN as a generator in extraction model, which can generate proposal regions of characters and speech balloons in an efficient way.

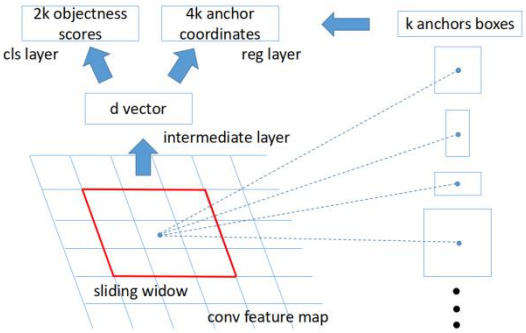


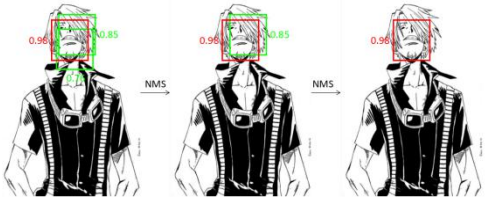
Fig.4 Region proposal network.

2) NMS(Non Maximum Suppression)

In object detection, the sliding window extracts the classification and score of each window (mapped box) by a classifier. However, many windows were extracted in duplicates. NMS is used to eliminate these duplicates. As the name, NMS implies, NMS[19] suppresses the smaller values and retain the bigger values in the local places. By this iteration, we can get the most accurate(the highest score) proposal region in this region.

As **Fig.5** shows. For example, if we have three proposal regions of face and each one of them has a score. Depending on the score, NMS will firstly find a lowest one in the proposal boxes of object region, which is 0.76. And then find the next lowest score in the rest of regions, which is 0.85. NMS will keep the bigger one and abandon the smaller one. So the 0.76 will be discarded. After many iterations, we can get the highest score(0.98 in figure5) in this region at last. That

is the most accurate candidate box.



@Eiichiro Oda

Fig.5 NMS algorithm.

3) Loss function

In order to compute the loss of model. We plan to assign a tag(0 or 1) to each candidate region. 0 means the area of region is not a object and 1 means it is a object. Also, assign a positive tag to a box whose IoU (Intersection-over-Union) (the ratio of intersection region and connection region) with an arbitrary box is larger than 0.7, and assign negative tags to other boxes.

According to the paper [9], the loss functions are defined as follows.

$$L = \frac{1}{N_{cls}} \sum_{i=0}^{n} L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_{i=0}^{n} p_i^* L_{reg}(t_i, t_i^*) \quad (1)$$

Function(1) is the images loss function. Function parameters, i is the mini batch(in paper,batch=256) anchor index. p is the "rpn_labels" part of the program (corresponding to anchor). p_i represents the probability that the candidate region is a target. When the candidate area is a positive tag, set $p_i^* = 1$. Otherwise $p_i^* = 0$. t_i is the predicted region coordinate vector (tx, ty, tw, th). t_i^* is the actual box coordinate vector.

Classification loss function(2) are log loss for two categories (target and non-target).

$$L_{cls}(p_i, p_i^*) = -\log[p_i p_i^* + (1-p_i)(1-p_i^*)] \quad (2)$$

Regression loss function(3) is as follows.

$$L_{reg}(t_i, t_i^*) = \sum_{i \in \{x, y, w, h\}} \text{smooth}_{L_1}(t_i, t_i^*) \quad (3)$$

The define of smooth is as follows.

$$\text{smooth}_{L_1} \begin{cases} 0.5x^2, & |x| < 1 \\ |x| - 0.5, & |x| \geq 1 \end{cases} \quad (4)$$

As function(5) shows, for regression loss, use four parameters to represent object position. (x,y) is the center coordinate of the prediction box. (x_a , y_a) is the coordinate of candidate box. (x^* , y^*) is the coordinate of the real area in metadata. w is the width of box and y is the height of box. N_{cls} and N_{reg} are the normalization parameters. According to the paper [20], N_{cls} and N_{reg} will set for 256 and 2400 in experiments. λ is the balance constant and will be set for 10.

$$\begin{cases} t_x = (x - x_a)/w_a, \; t_y = (y - y_a)/h_a \\ t_w = \log(w/w_a), \; t_h = \log(h/h_a) \\ t_x^* = (x^* - x_a)/w_a, \; t_y^* = (y^* - y_a)/h_a \\ t_w^* = \log(w^*/w_a), \; t_h^* = \log(h^*/h_a) \end{cases} \quad (5)$$

D. ResNet

There is no CNN architecture deeper than the ILSVRC 2015 champion, Microsoft's ResNet. The ResNet not only created a new record on the number of layers, but also reached a staggering error rate, 3.6%(human recognition level is about 5% to 10%).

The motivation for creating ResNet is the so-called "degradation" problem, which is that when the level of the model become deeper, the

error rate increases. However, according to the view in paper[17], the depth of the model becomes deeper and the learning ability should be enhanced. Therefore, a deeper model should not produce a higher error rate than a shallower model. The reason for this “degradation” is the optimization problem. When the model becomes complicated, the optimization of GD(gradient descent) becomes more difficult, resulting that the model fails to achieve a good learning effect. In order to solve this problem, Shaoqing Ren et al.[17] proposed the ResNet.

A major contribution of ResNet is to solve the problem of the gradient disappearing. Gradient disappearance or gradient explosion has always been a major issue when people train a deep network. Even with the ReLU function, localized normalization methods and other methods, the model cannot get a good performance in learning features when it has a deeper conv layers(more than30). Gradient values will inevitably become smaller and smaller as they propagate to the first few layers. As **Fig.6** shows, In Resnet, an identity mapping is added in every two layers, so that the data can be directly transmitted without hindrance. In this way, gradient disappearance problem has been solved. The thought of Resnet is that if we have a good network A. Then, design a network B, which is mapped from A by an identity mapping. At least net B can get the same performance as A net and would not be even worse.

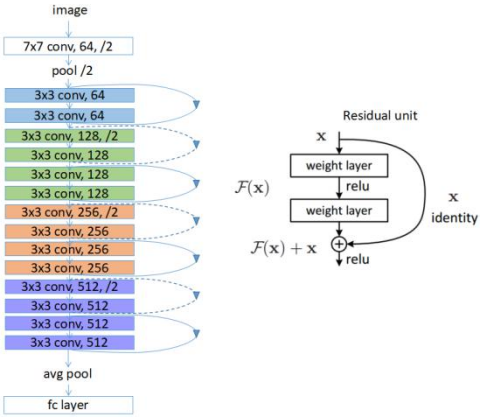


Fig.6 ResNet structure. From paper[17].

In extraction model, especially the data is black-and-white manga, we need that the network can extract abstract features effectively. Because the color information of manga that we can use is limited. So replacing VGG model, we choose Resnet as the CNN structure.

IV. Research Method

A. Framework and database

The framework will use the open source, Faster r-cnn with the model Resnet50 model (https://github.com/keras-team/keras/blob/master/keras/applications/resnet50.py).The platform we plan to use is the Tensorflow of Google. We will choose Manga109[21] as the training and testing database. Manga109 is a database that can provide metadata of manga. It contains109 Japanese comic books, for a total of 21,142 pages. Each comic book contains 194 pages on average. The comics were created by 94 professional creators. The dataset covers the era from 1970 to 2010 and includes various genres (e.g., science fiction, humor, military, and love and romance) [21]. The dataset in Manga109 have already been labeled for doing deep learning research. The labeled characters, face and speech balloons in **Fig.7. Also, w**e choose Manga109 as our database, since the extraction model is a supervised learning model. Moreover, we plan not only to use Manga109 to train the model, but also add more metadata into Manga109 to expand its scale.

@Ken Akamatsu

Fig.7 Manga109 data samples. From paper[21].

B. Training method

1) Hard example mining

As **Fig.8** shows, sometimes the character in the whole image is much smaller than the background area, so the negative space of sample is very large. If such data were taken directly to train the model, the model would likely tend to divide all samples into negative samples, resulting in difficulty converging. Using the Hard example mining method[22] in training can solve this problem. Firstly, collect all detected boxes from negative sample image (no character or speech balloon) in the first training. Then use this negative model to form a error sample set. Add this sample set to the training set to train new ones.

@kobayashi yuki

Fig.8 The example of hard example.

2) Train in multi-scale

In actual scenes, there are a lot difference between the far and near targets. In training, multi-scale training is used to improve the robust ability of model. As **Fig.9** shows, Firstly, we set a variety of scales for a picture. Then randomly select the scale of images for training. Therefore, the trained model can learn features in various sizes. Through experiments, it is proved that the using of multi-scale training can make the object size in an average distribution, so that the trained model is more robust[23] to the object size.

However, this way also has a demerit, which is that it will cost a lot time to train the network by three times the amount of images. So we will consider to replace this method with FPN in late period.

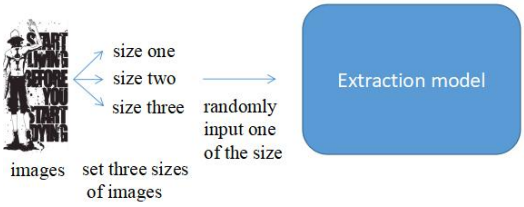


Fig.9 The flow of train in multi-scale.

C. Training

The flow of training and testing is as **Fig.10** shows. In training, in order to fit model, firstly do processing of Manga109 database. Secondly load the pre-trained model Resnet50 (https://github.com/fchollet/deep-learning-models/releases). Thirdly, train the network in an usual way and test the network. Depends on test results, select hard example mining method or multi-scale method to train model repeatedly. At last,, tune network.

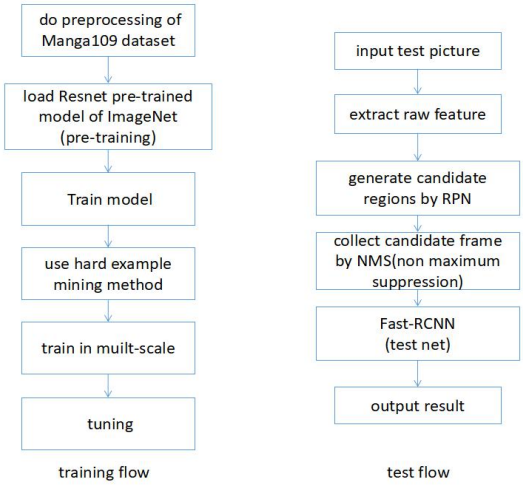


Fig.10 Training and testing flow.

D. Experiment and analyze

Use different methods (plan to use FCN, SSD algorithm and VGG net) to implement extraction. Evaluate the accuracy of only detecting characters, only detecting speech balloons and both detecting characters and speech balloons. Evaluate the true positive (TP), false negative(FN), false positive(FP), recall,

precision and the average precision of each component's extraction. Finish the PR curves. Compare with other two methods.

V. Research Plan

The aim of this research proposal is thorough combining the manga with deep learning to complete a model that can automatically extract components of manga more efficiently. Thanks to the prior research, I can have good comprehension of a lot of different models, which are used in past. Basing on that, I plan to use different network and training method to implement a new model, theoretically which will have a better performance in extracting components.

This research also has an exciting prospect. For we can use the extraction model as the basic function to implement many new services for e-comic market, such as searching by pictures, automatically changing the size of speech balloons(like Bubble Zoom of Google Play app), editing e-comics on line and so on. All of these functions are not been applied in Japanese e-comics market yet. As far as I am concerned, it has great potential for development.

I plan to use four steps to complete the research.

First step. In order to make sure that I can totally comprehend the theories of deep leanings(maybe there are some mistakes in my plan now), I plan to read more prior researches and learn more the basic knowledge of deep learning(programming and math). Moreover, I will try to understand more algorithms and networks models, compare the merits and demerits of different networks.Then, decide which algorithms and network will be used in model. Simultaneously, I plan to supplement the database of Manga109 in this year, so as to we can have more effective training samples in experiment.

Second step. Finish the code of model and test debug. Then start train and test. In training and testing, collect the valid data, such as true positive (TP), false negative(FN), false positive(FP), recall, precision and the average precision of each component's extraction. In order to implement the extraction model by different methods, learn different algorithms and network in advance. Do the preparations of different methods' coding and testing.

Third step. Finish the coding of different methods(plan to use FCN, SSD algorithm and densenet) and test. Compare the performance of different methods. Try to improve the model with other efficient technology.

Fourth step. Write all the research results into paper.

During my time as a graduate student, I plan to complete the first step. In addition, in this one year I will do enough preparation for the graduate entrance exam and proposing paper.

VI. References

[1] 植村八潮. 電子書籍がもたらす出版・図書館・著作権の変化 現状分析と今後のあり方の検討[J]. 情報管理 2013, 56(7): 403-413.

[2] 落合早苗. 電子書籍ビジネス調査報告書 2015 [M]. インプレス, 2018.

[3] 植村八潮. “電子書籍” の市場拡大と概念拡張［J］. 情報の科学と技術, 2017, 67(1): 2-7.

[4] 鈴木崇宏, 高橋光輝. 電子書籍における漫画の動向−電子書籍販売の市場動向についての調査報告[J]. 研究報告コンシューマ・デバイス & システム (CDS), 2015, 2015(49): 1-8.

[5] Tanaka T, Toyama F, Miyamichi J, et al. Detection and classification of speech balloons in comic images[J]. The journal of the Institute of Image Information and Television Engineers, 2010, 64(12): 1933-1939.

[6] Arai K, Tolle H. Method for real time text extraction of digital manga comic[J]. International Journal of Image Processing (IJIP), 2011, 4(6): 669-676.

[7] Matsui Y, Ito K, Aramaki Y, et al. Sketch-based manga retrieval using manga109 dataset[J]. Multimed

ia Tools and Applications, 2017, 76(20): 21811-21838.

[8] Tanaka T, Toyama F, Miyamichi J, et al. Detection and classification of speech balloons in comic images[J]. The journal of the Institute of Image Information and Television Engineers, 2010, 64(12): 1933-1939.

[9] Chu W T, Li W W. Manga FaceNet: Face Detection in Manga based on Deep Neural Network[C]//Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval. ACM, 2017: 412-415.

[10] LeCun Y, Boser B E, Denker J S, et al. Handwritten digit recognition with a back-propagation network[C]//Advances in neural information processing systems. 1990: 396-404.

[11] LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[12] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.

[13] Girshick R. Fast r-cnn[J]. arXiv preprint arXiv:1504.08083, 2015.

[14] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.

[15] Uijlings J R R, Van De Sande K E A, Gevers T, et al. Selective search for object recognition[J]. International journal of computer vision, 2013, 104(2): 154-171.

[16] Zitnick C L, Dollár P. Edge boxes: Locating object proposals from edges[C]//European Conference on Computer Vision. Springer, Cham, 2014: 391-405.

[17] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

[18] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 3431-3440.

[19] Zelnik-Manor L, Perona P. Self-tuning spectral clustering[C]//Advances in neural information processing systems. 2005: 1601-1608.

[20] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[C]//Advances in neural information processing systems. 2015: 91-99.

[21] Fujimoto A, Ogawa T, Yamamoto K, et al. Manga109 dataset and creation of metadata[C]//Proceedings of the 1st International Workshop on coMics ANalysis, Processing and Understanding. ACM, 2016: 2.

[22] Shrivastava A, Gupta A, Girshick R. Training region-based object detectors with online hard example mining[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 761-769.

[22] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.