Publications

Proceedings

[1] Chi W, **Feng X(*euqal contribution),** Chen Y, et al. Stress Prediction Based on Multi-Granularity Linguistic Knowledge. IEEE International Conference on Acoustics, Speech and Signal Processing International Conference on Pattern Recognition and Machine Learning, 2023(ICASSP'23), 5 pages. reviewing Github

[2] Wang Z, **Feng X**, Tang J, et al. Deep Knowledge Tracing with Side Information.
International conference on artificial intelligence in education. Springer, Cham, 2019(AIED'19), 5 pages. PDF

[3] **Feng X**, Xie R, Sheng J, et al. Population statistics algorithm based on MobileNet. Journal of Physics: Conference Series. IOP Publishing,2019(ICSP'19), 6 pages. PDF

[4] Zhang J, **Feng X**, Chen Y, et al. Prosody Prediction with Discriminative Representation Method.
International Conference on Pattern Recognition and Machine Learning, 2022(PRML'22), 5 pages. PDF

[5] Rong Xie, **Feng X**. A method of quick edge detection based on Zynq.
International Conference on Cloud Computing and Internet of Things, 2018(CCIOT'18), 5 pages. PDF

[6] Sheng J, **Feng X.** Research on the Internet of Things Platform for Smart and Environmental Protection.
International Conference on Cloud Computing and Intelligence Systems, 2018(CCIS'18), 5 pages. PDF

Patents

• FENG XIAOQIN, LEI XIN, LI ZHIFEI. Polyphone labeling method and device, and computer readable storage medium.
Mobvoi(algorithm),2019,CN111078898A

• FENG XIAOQIN, LEI XIN, LI ZHIFEI. Speech synthesis method and device and computer readable storage medium.
Mobvoi(algorithm),2020,CN110970013A

• FENG XIAOQIN, LI NA, LEI XIN, LI ZHIFEI. Polyphone labeling method and device, and computer readable storage medium.
Mobvoi(application),2020,CN111145724A

• other 2 co-patents published: CN111079428A CN111178042A

Activities

"Movie Recommendation Based on Knowledge Graph" PDF
Aug, 2018 Peking University & Sinovation Ventures

Theses

• Xiaoqin Feng. 2019. Research on multi-scene video intelligent processing system and scheduling management algorithm.
In the Institute of Software Engineering. Beijing University of Technology. 78 pages. Master Thesis. PDF

• Xiaoqin Feng. 2016. Intelligent Laboratory Management System.
In the Institute of Computer Science and Engineering. Southwest Minzu University. 37 pages. Bachelor Thesis.

Deep Knowledge Tracing with Side Information

Zhiwei Wang*[1], Xiaoqin Feng*[2], Jiliang Tang[1], Gale Yan Huang[2], and Zitao Liu**[2]

[1] Data Science and Engineering Lab, Michigan State University, USA
{wangzh65,tangjili}@msu.edu
[2] TAL AI Lab, Beijing, China
{fengxqin,galehuang,liuzitao}@100tal.com

Abstract. Monitoring student knowledge states or skill acquisition levels known as knowledge tracing, is a fundamental part of intelligent tutoring systems. Despite its inherent challenges, recent deep neural networks based knowledge tracing models have achieved great success, which is largely from models' ability to learn sequential dependencies of questions in student exercise data. However, in addition to sequential information, questions inherently exhibit side relations, which can enrich our understandings about student knowledge states and has great potentials to advance knowledge tracing. Thus, in this paper, we exploit side relations to improve knowledge tracing and design a novel framework DTKS. The experimental results on real education data validate the effectiveness of the proposed framework and demonstrate the importance of side information in knowledge tracing.

1 Introduction

Knowledge tracing - where machine monitors students' knowledge states and their skill acquisition levels - is essential for personalized education and a fundamental part of intelligent tutoring systems [15,7,1,12]. However, tracing student knowledge states is inherently challenging because of the complexity of human learning process, which involves a variety of factors from diverse domains such as neural science [3,4], psychology [10], and education [8]. Meanwhile, the large amount of data produced by a growing number of online education platforms and recent advances of machine learning technology provide us with unprecedented opportunities to build advanced models for accurate knowledge tracing. Consequently, it has garnered widespread attention from researchers in both education and artificial intelligence communities [12,16,14]. Recently, one framework named Deep Knowledge Tracing (DKT) that is based on deep neural networks has shown superior performance over previously proposed knowledge tracing models [12]. Specifically, based on student historical answered questions, it is able to predict student performance on future questions with high accuracy. The key reason of the success of DKT is its ability to capture the sequential dependencies among questions embedded in the question answer sequences.

In fact, in addition to the sequential dependencies, questions naturally exhibit side relations due to their intrinsic properties. For example, questions are

* Work was done when the authors did internship in TAL AI Lab
** Corresponding Author: Zitao Liu

typically designed to improve certain concepts or skills. Thus, questions with similar underlying concepts or skills are inherently related. These relations can be represented as a question-question graph where nodes are questions and an edge exists in two questions if they are designed to examine similar sets of skills and concepts. The question-question graph provides rich information that can lead us to a better understanding of student knowledge states and exploiting such information has the great potential to improve the knowledge tacking performance.

In this work, we exploit question relation information for better knowledge tracing and propose a framework DTKS that can capture both sequential dependencies and intrinsic relations of questions simultaneously. In summary, the contributions of this work are: 1) We identify the importance to incorporate side relations of questions into knowledge tracing; 2) We design a novel framework DKTS that provides a principled approach to capture both sequential and side relation information to model the student knowledge states and accurately predict their performance; and 3) We demonstrate the effectiveness of the proposed framework with real data.

2 Related Work

In this section, we briefly review the related works. Knowledge tracing is a long established research question and an essential task for computer assisted education [15,2,7,1,12,16]. Previously, Bayesian Knowledge Tracing (BKT) based approach has been in predominate use [15,1]. It represents the student knowledge state with a set of binary variables and each variable corresponds to student understanding of a single concept [7]. Other approaches such as Learning Factors Analysis [5] and ensemble methods [2] have also been proposed and achieved comparable performance with BKT. Recently, deep neural network based approach has become increasingly popular [12,16]. Models in this line such as DKT [12] represent student knowledge state with continuous and expressive latent vectors and are able to capture the complexity of knowledge state. However, few of them incorporates the question relation information, which could be very helpful for knowledge tracing tasks.

3 The Proposed Framework

In this section, we introduce our proposed model DKTS that is able to incorporate question relations in modeling student knowledge state. The overall structure of the proposed model is shown in Figure (1). Before detailing each layer next, we first introduce the notations. Vectors and matrices are represented with bold lower-case letters such as $\mathbf{h}$ and bold upper-case letters such as $\mathbf{W}$. In addition, the i^{th} entry of vector $\mathbf{h}$ is denoted as $\mathbf{h}(i)$ and the entry at the i^{th} row and j^{th} column of matrix $\mathbf{W}$ as $\mathbf{W}(i, j)$.

Input and Embedding Layers: The input of the framework is the student past question answer sequence $S = (x_1, x_2, \cdots, x_n)$, where $x_j = (q_j, a_j)$ involves a question q_j and the correctness of the student answer denoted as $a_j \in \{0, 1\}$. We represent x_j as $\mathbf{x}_j$ using an embedding layer.

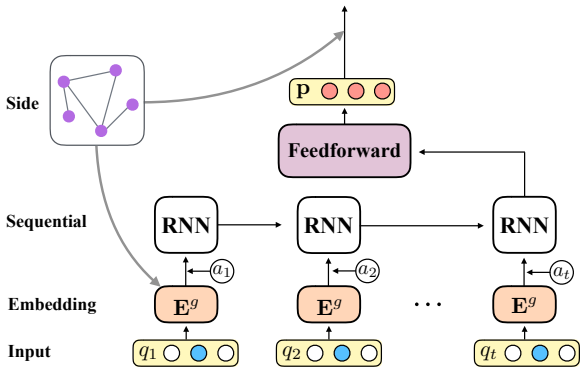

Fig. 1: The network architecture of the proposed framework.

The Sequential Layer: We take advantage of RNN models to trace student knowledge states. Specifically, at time step t, RNN maintains a latent vector $\mathbf{h}_t \in \mathbb{R}^{n_h}$ representing student knowledge state through the following cell structure: $\mathbf{h}_t = tanh(\mathbf{W}\mathbf{x}_t + \mathbf{U}\mathbf{h}_{t-1} + \mathbf{b})$. Thus, this recursive structure naturally describes the evolution of student knowledge state $\mathbf{h}_t$ that is driven by the previous knowledge state $\mathbf{h}_{t-1}$ and current observation $\mathbf{x}_t$. In practice, more advanced recurrent cells such as long short-term memory unit (LSTM) and gated recurrent unit cell (GRU) [11,6] often achieve better performance than original cell. We investigate both of them in this work. After sequential layer, we design a feedforward layer to predict the student future's response to each question based on the final knowledge state representation $\mathbf{h}$ by following equation: $\mathbf{p} = \sigma(\mathbf{h}\mathbf{W}^p + \mathbf{b}^p)$, where $\mathbf{p}(i)$ indicates the probability that the student can answer the i^{th} question correctly.

The Side Layer: In side layer, two model components are designed to capture the question relation. Firstly, instead of using embedding layers, we apply graph embedding algorithms such as LINE [13] and Node2Vec [9] to the question-question relation graphs to obtain the question representations that preserve the question relations. Secondly, to impose the intuition that if a pair of questions (e.g., i^{th} and j^{th} questions) requires similar skills or involves similar concepts, the probability for a given student answering the two questions correctly should also be similar, we design the following regularization term $\mathcal{L}_r = \frac{1}{2}\mathbf{p}^T\mathbf{L}\mathbf{p}$, where $\mathbf{L}$ is the Laplacian matrix of adjacent matrix $\mathbf{A}$ representing the question relation graph.

The Loss Function: With the prediction $\mathbf{p}$ obtained from sequential layer and the relation regularization term $\mathcal{L}_r$, we define the loss function of the proposed framework DKTS for each training data as $\mathcal{L} = \mathcal{L}_p + \alpha\mathcal{L}_r$, where α is adopted to control the contribution of relation regularizer and $\mathcal{L}_p$ is the binary cross-entry loss that is defined as:

$$\mathcal{L}_p = -a_{t+1}\log(\mathbf{p}^T\mathbf{q}_{t+1}) - (1 - a_{t+1})log(1 - \mathbf{p}^T\mathbf{q}_{t+1}) \tag{1}$$

where $\mathbf{q}_{t+1}$ is the one-hot encoding of the question at time step $t+1$.

4 Experiment

In this section, we conduct experiments on real education data to verify the effectiveness of the proposed model.

Dataset: We collect a student question answer behavior dataset from one of the most popular GMAT preparation mobile applications in China. It contains 8,684 questions and 90831 anonymized students and is cleaned by a filtering process. For each student, we collect her question answer behaviors and form a sequence of behaviors ordering by time information. A question relation graph is constructed according to the underlying knowledge and skills.

Baselines: In baselines, we use RNN, LSTM, and GRU to model the students knowledge state and represent question by embedding vectors that are sampled from a Gaussian distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$ (Gaussian), or obtained through graph embedding algorithms (LINE, Node2Vec). Note that previously proposed DKT model uses LSTM to learn student knowledege state with question representation vectors sampled from Gaussian distribution [12].

Experimental Results: We evaluate the prediction performance by area under the curve (AUC) and a higher AUC indicates better performance. The results are shown in Table 1. We observe that 1) The embedding vectors that preserve the question relation information significantly improve the prediction performance, which clearly demonstrates the importance of question relation information for knowledge tracing tasks; and 2) The proposed framework DKTS outperforms all other methods by a large margin. We contribute the superior performance of the proposed model to its ability to incorporate question relation information.

Table 1: Performance Comparison Results. 'NA' indicates not applicable.

Method	Question Embedding		
	Gaussian	LINE	Node2Vec
RNN	0.6527	0.7015	0.6988
LSTM	0.6999	0.7152	0.7140
GRU	0.7074	0.7173	0.7165
DKTS	NA	0.7338	0.7340

5 Conclusion

In this work, we exploit question relation information for knowledge tracing tasks. Specifically, we design a novel deep neural network based framework that is able to capture the sequential dependencies and intrinsic relations of questions to trace the student knowledge state. Moreover, we evaluate the proposed framework with real education data on student future interaction prediction task. The experimental results have clearly demonstrated the importance of the question relation information and the proposed framework outperforms state-of-the-art baselines significantly.

Acknowledgements. Zhiwei Wang and Jiliang Tang are supported by the National Science Foundation (NSF) under grant numbers IIS-1714741, IIS-1715940, IIS-1845081 and CNS-1815636, and a grant from Criteo Faculty Research Award.

References

1. d Baker, R.S., Corbett, A.T., Aleven, V.: More accurate student modeling through contextual estimation of slip and guess probabilities in bayesian knowledge tracing. In: Intelligent Tutoring Systems (2008)
2. d Baker, R.S., Pardos, Z.A., Nooraei, B.B., Heffernan, N.T.: Ensembling predictions of student knowledge within intelligent tutoring systems. In: UMAP (2011)
3. Bassett, D.S., Porter, M.A., Mucha, P.J., Carlson, J.M., Grafton, S.T.: Dynamic reconfiguration of human brain networks during learning. PNAS (2011)
4. Caine, R.N., Caine, G.: Understanding a brain-based approach to learning and teaching. Educational Leadership **48**(2), 66–70 (1990)
5. Cen, H., Koedinger, K., Junker, B.: Learning factors analysis–a general method for cognitive model evaluation and improvement. In: ITS. Springer (2006)
6. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 (2014)
7. Corbett, A.T., Anderson, J.R.: Knowledge tracing: Modeling the acquisition of procedural knowledge. User modeling and user-adapted interaction (1994)
8. Felder, R.M., Silverman, L.K., et al.: Learning and teaching styles in engineering education. Engineering education **78**(7), 674–681 (1988)
9. Grover, A., Leskovec, J.: node2vec: Scalable feature learning for networks. In: KDD (2016)
10. Hilgard, E.R.: Theories of learning. (1948)
11. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation (1997)
12. Piech, C., Bassen, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L.J., Sohl-Dickstein, J.: Deep knowledge tracing. In: NIPS (2015)
13. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: Line: Large-scale information network embedding. In: WWW (2015)
14. Wang, L., Sy, A., Liu, L., Piech, C.: Deep knowledge tracing on programming exercises. In: L@S (2017)
15. Yudelson, M.V., Koedinger, K.R., Gordon, G.J.: Individualized bayesian knowledge tracing models. In: AIED (2013)
16. Zhang, J., Shi, X., King, I., Yeung, D.Y.: Dynamic key-value memory networks for knowledge tracing. In: WWW (2017)

Population Statistics Algorithm Based on MobileNet

IOP Conf. Series: Journal of Physics: Conf. Series **1237** (2019) 022045 doi:10.1088/1742-6596/1237/2/022045

Population Statistics Algorithm Based on MobileNet

Xiaoqin Feng, Rong Xie, Junyang Sheng, Shuo Zhang

Beijing Engineering Research Center for IoT Software and Systems, Beijing University of Technology, Beijing 100124, China

Email: fengxqin@emails.bjut.edu.cn

Abstract. In today's society, intelligent video surveillance plays an important role in social security, traffic scheduling, national security and other fields. One of the research hotspots is people statistics based on image processing, which has strategic significance in practical applications. Aimed at the problem that the low accuracy in the actual application scenario, the limited hardware resources, and the low operation efficiency, this paper proposes a multi-feature target detection model based on the lightweight deep learning network MobileNet [1], which can be used in intelligent terminals. The basic feature-extraction network MobileNet as a lightweight network can provide a flexible alternative configuration in terms of efficiency and accuracy. The underlying detection network selects a single deep nerual network, named SSD [2]. The algorithm can achieve multi-scale target detection, and uses the target position and category to perform one-time regression. In this paper, the activation function of SSD is changed into SeLU (scaled exponential linear units) [3], which improves the robustness of the algorithm. At the same time, the work of sample diversity and data enhancement has been made, and the characteristics of the human body above the shoulders have been fully utilized. Experiments have shown that the improved network structure based on MobileNet has higher detection accuracy, lower delay, excellent robustness, while the number of model parameters is effectively reduced.

1. Introduction

With the development of economy and society, the population of China has an exponential trend, and the statistical research on the number of people in various fields of life has become a realistic demand. At the same time, with the continuous development of computer vision technology, image processing technology has broad application prospects and development space in this field. However, in practical application scenarios, due to target occlusion, contour non-rigidity, attitude variability, image resolution, ambient illumination and complex environment, etc., it poses a huge challenge to accurate population statistics. In addition, due to the limited resources of embedded intelligent terminals, the computational complexity of image algorithms based on deep learning makes the application of algorithms a major research problem. Designing a simple, effective, robust, real-time, and accurate population statistics algorithm has become a challenge.

At present, the population statistics algorithm combines multiple techniques such as background analysis, feature extraction, object detection, target segmentation, and target tracking. According to different means, it is mainly divided into two categories: 1) Indirect method, also known as feature-based method, refers to establishing a functional relationship between target characteristics and number of people to perform metric calculation. Hashemzadeh M [5]et al. proposed population density and population occlusion characteristics based on key points and foreground segmentation to estimate the number of people; Chang QL [6] et al proposed to calculate the population occlusion factor by

normalizing the foreground and corner information to pass the two features through backward propagation(BP). The network completes the number of people, etc. The above method is mainly applied to scenes where pedestrians are highly dense, by extracting features in the image and establishing an estimated function statistic. 2) Direct method: The core idea is to extract the target features, learn a classification model to achieve target detection, and the classic pedestrian detection model of HOG+SVM proposed by Navneet Dala [7]. The classification model of SVM is established by extracting the HOG (gradient histogram feature) feature of the image; similarly, the Haar-like+Adaboost cascade classification model proposed by Paul Viola [8]. In addition, the convolutional feature based on deep learning combined with traditional machine learning classification has become the current mainstream method, with high precision and strong generalization ability [4].

At present, people believe that the shortcomings in this aspect are in three parts [9]: 1) Shallow learning. The accuracy is not up to the requirement. Although a better method is proposed, such as the DPM algorithm based on the human body component, it has a better detection effect in a complicated environment. However, because of the existence of multiple classification models, the efficiency is difficult to improve [10] 2) Deep learning. Based on various CNN algorithms, it is mainly divided into one-stage and two-stage. Typical examples include RCNN, Faster-RCNN, SSD, etc. Because of their strong feature expression ability, these algorithms can achieve a better detection results and strong generalization ability in combination with traditional machine learning classifiers. However, the depth of its model and the huge amount of computation make it difficult to use in real life. 3) Data labeling problem, currently there is no standard data set for the population statistics application scenario, such as the PASCAL-VOC data set, the standard is the whole body. The coverage area is relatively wide and there are other types of coverage. This paper mainly constructs a comprehensive human head detection data set, based on the current popular lightweight network MobileNet to achieve image feature extraction. The lower layer adopts the SSD single detection model and adds the activation function SeLU. Experiments show that the improved network structure based on MobileNet has higher detection accuracy, lower latency, better robustness, and the number of model parameters is effectively reduced.

2. Model

This paper selects MobileNet as the feature extraction network and SSD as the underlying target detection framework. There are two versions of MobileNet, and the activation function of the overall model structure change into SeLU function, which improves the robustness of the algorithm

2.1. MobileNet:Feature extraction network

MobileNet, mainly a lightweight deep neural network proposed by Google to solve the problem that mobile embedded terminals cannot be applied. Its research direction lies in the aspect of model compression, and its core idea is the ingenious decomposition of convolution kernel. It can effectively reduce network parameters while taking into account optimization delay.

2.1.1. MobileNetV1

The network design is based on a streamlined architecture that uses a deeply separable convolution to build a lightweight deep neural network. Introducing two global hyperparameters: the width parameter and the resolution parameter, effectively balancing the efficiency and accuracy. These two parameters allow us to choose the right model for our application based on actual problems.

Depthwise convolution applies convolution kernels to each channel, the 1*1 pointwise convolution is used to combine the output of channel convolutions. This kind of idea can achieve the same effect as traditional convolution, but it can effectively reduce the size of the model. Fig 1, 2 show a comparison of standard convolution and depthwise separable convolution.

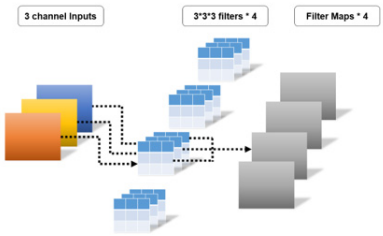

Fig.1 Standard Convolution: The three-channel RGB image input as M*M passes through the convolution process of 3*3*3*4 convolution kernel (eg: input channel: 3, output channel: 4), and finally outputs 4 feature maps. The specific size also refers to the padding and strides parameters

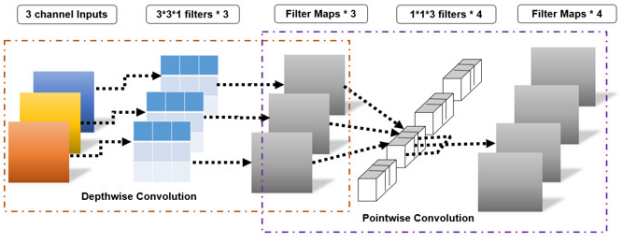

Fig.2 Depthwise Separable Convolution: Decompose a standard convolution operation into two steps, namely Depthwise Convolution and Pointwise Convolution.

It can be found that, unlike traditional convolution operations, Depthwise Convolution performs a separate convolution for each channel, that is, a channel is convolved by only one convolution kernel. The number of Feature maps after depth convolution is the same as the channel of input layer, and the feature maps cannot be extended. And this operation performs convolution operation independently for each channel of the input layer, and cannot effectively utilize the feature information of different channels in the same spatial position. Therefore, Pointwise Convolution is required to combine these feature maps to generate a new feature map. In this separation operation, the amount of calculation is greatly reduced, and the calculation ratio (CCR=DW Conv Cost / Std Conv Cost) of the two is calculated by the following formula (width hyper-parameter and resolution hyper-parameter are omitted)

$$\mathrm{CCR} = \frac{G_k^2 * M * G_f^2 + M * N * G_f^2}{G_k^2 * M * N * G_f^2}$$
$$\mathrm{CCR} = 1/N + 1/G_k^2 \quad (1)$$

Where N is the channel of feature maps, usually larger, usually greater than 10. G_k^2 is the size of the convolution kernel, typically 3*3, and the ratio is a number less than one.Therefore, the DW convolution reduces the amount of computation compared to the Standard convolution. In addition, the formula for calculating the network after adding two hyper-parameters is as follows:

$$G_k^2 * \alpha M * \rho G_f^2 + \alpha M * \alpha N * \rho G_f^2 \quad (2)$$

Where α is the width parameter, the role is to change the input and output channels, reduce the number of feature maps, the value is 0~1; ρ is the resolution parameter, the role is to change the resolution of the input layer. The combination of the two can further reduce the amount of calculation.

2.1.2. MobileNetV2

On the basis of MobileNetV1, an upgraded version is proposed, which ensures the accuracy of the model and reduces the amount of calculation [11].

Two basic structures are proposed: 1) Linear Bottlenecks, which replaces the ReLU layer with a linear transformation layer to reduce information loss. 2) Inverted Residual Structure, which uses a shortcut to link the block's input and output (element-wise). Compared to the original residual structure, this reverses the order of transformation of the internal data dimensions, which saves memory and enriches feature information. Fig 3 shows the comparison between the original residual structure and the Invert residual structure:

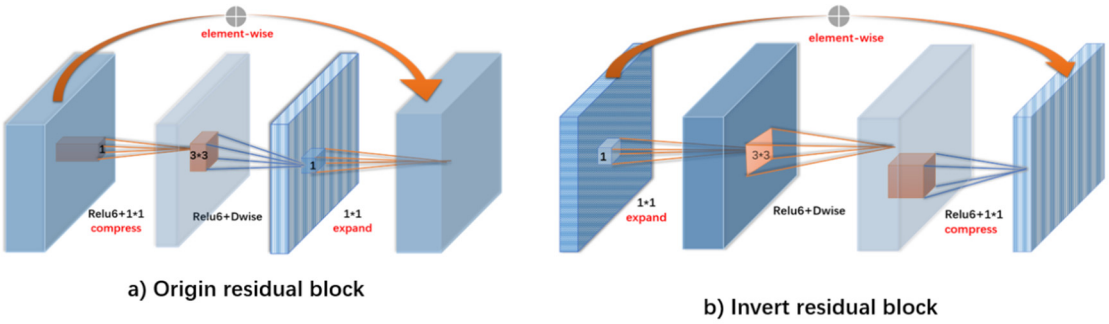

Fig.3 Comparing the original RBlock, MobileNetV2: 1) 'expand' the channel of the feature map after a 1*1 Conv layer; 2) extract the feature information via the 3*3 Conv layer; 3) after a 1*1 Conv Layer, 'compresses' the feature map channel back to its original size. The original is to first "compress" and then "expand"

2.2. Improved SSD Structure

SSD is a single-shot detection model. The core idea is to obtain the location and category of the target in a regression manner on the multi-scale feature map. However, there is a problem of inaccurate detection of small targets. Through analysis, the factors affecting the detection of small targets are mainly the resolution of the feature map and the global information and feature extraction capabilities. The residual block structure of MobileNetV2 can improve the high-resolution low-level feature expression of the feature map used for SSD detection. In addition, the original SSD framework uses the ReLU activation function, which is modified into a SeLU activation function. The network obtained by the activation function has self-normalization characteristics. Among them, ShaoHua. et al. compared SeLU with other activation functions, which proved its effectiveness and robustness, even surpassing batch normalization.

Comparison of ReLU & SeLU: People know that the addition of a neuron to an activation function has the ability of non-linear representation, which is the biggest difference between neural networks and linear classifiers. Compared to the original activation function Sigmoid, ReLU has three differences: 1) unilateral suppression 2) wider excitation boundary 3) sparse activation. ReLU can transfer gradients very well: after repeated back propagation, the gradient will not be greatly reduced, suitable for training deep neural networks. However, experiments have shown that the ReLU activation function is prone to training interruptions, and ReLU-enforced sparse processing reduces the effective parameter capacity of the model (when x < 0, the negative gradient is zeroed and will no longer be activated by any data. It is called neuron "necrosis"). One of the similarities between ReLU and Sigmod is that the results are all positive, reducing the ability to express features

In 2017, the literature [24] introduced a new activation function SeLU (scaled exponential linear units), introducing the properties of self-normalization. The following are the mathematical formulas for the SeLU and ReLU:

$$ReLU_{(z)} = \begin{cases} z & z > 0 \\ 0 & z \leq 0 \end{cases} \tag{3}$$

$$SeLU_{(z)} = \gamma \begin{cases} z & z > 0 \\ \alpha(exp(z) - 1) & z \leq 0 \end{cases} \tag{4}$$

$$where\ \gamma \approx 1.0507,\ \ \alpha \approx 1.673$$

SeLU mainly uses a function F to establish the mapping relationship of the neural network layer. At the same time, the parameters are transformed to a fixed mean and variance to achieve the

normalized effect. Compared with ReLU, it has the following advantages: 1) Strong convergence property, even if there is noise and interference in the data, it will converge faster after multi-layer forward and backward propagation; 2) Regularization effect, enhance Algorithm robustness; 3) In addition, for the excitation values that do not approximate the unit variance, the variance has upper bound and lower bound, so gradient disappearance and gradient explosion are almost impossible.

Model Structure:This article selects the SSD detection framework based on MobileNet. Adding 8 convolution layers behind the conv13 of MobileNet. A total of 6 layers are extracted for detection, and the activation function is SeLU. Fig4 is the model structure diagram of this paper.

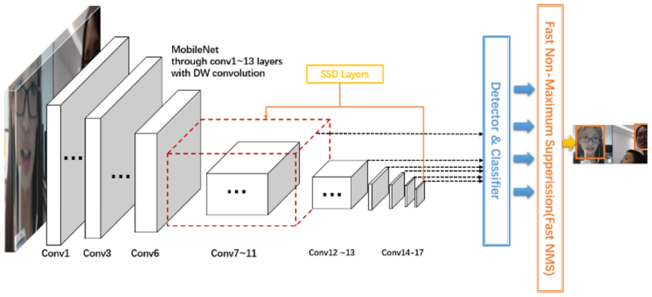

Fig.4 Model structure, the left part is the feature extraction network MobileNet, which contains the output of the conv1-conv13 layer. After the conv13, 8 layers of convolution are added, and 4 layers and conv11 and conv13 of MobileNet are selected as the scale feature map of the SSD, and finally attributed to a prediction unit. The activation function in the middle uses the SeLU function.

3. Experiments

In this section, introducing the experimental operating environment, data set, evaluation criteria, and training process in turn. Finally, the effects of different models on the same data set are compared.

3.1. Experimental Preparation

TABLE I. DATASETS

DataSet	***Labels***	***Datasize***	***Pixelsize***
Pascal VOC 2007-06	21	9963	500*375
Pascal VOC 2012-11	21	17125	500*375
INRIA	2	2416	96*160
MyData	2	5000	300*375

The experimental hardware environment is Ubuntu16.04+cuda8.0+GTX1080. Using Keras as a deep learning framework to define different models, and the back end is a Tensorflow implementation that supports GPU computing. The data set is shown in Table I. In order to construct the omnidirectional data set of the human head, the INRIA pedestrian detection data set and the PASCAL-VOC 2007+2012 database are used. However, the data that meets the requirements is still relatively small. Therefore, a large number of image data containing the target (person) are captured online and relabeled using the LabelImg tool and the PASCAL-VOC labeling format was used. In order to get all-round and feature-rich data, not only focus on the facial skin when marking, and the features above the shoulder will be added to the target frame. The PASCAL-VOC data has also been recalibrated to obtain richer and more accurate feature information. In addition, over-fitting of data information is avoided, and the generalization ability of the model is increased. The code and dataset will be published later on github.

In order to enhance the complexity of the data, series of data enhancement work was putted into the model, such as random clipping, brightness transformation, etc.

3.2. Evaluation Criterion

In this paper, the target detection model is selected, and people counting in different scenarios is applied. RMSE is selected as the evaluation criterion. The calculation formula is:

$$RMSE(X,f) = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(f(x^i) - y^{(i)})^2} \quad (5)$$

Where $\boldsymbol{y^{(i)}}$ is the number of original people in each picture, $\boldsymbol{f(x^i)}$ is the number of people detected, and m is the number of video frames detected. In addition, this article uses the frames per second (FPS) to measure the target detection speed and uses 25 fps as the real-time consideration threshold.

3.3. Experimental Results and Analysis

In order to compare the network parameters, detection effect and real-time performance, the popular VGG convolution network is choosed, which is more popular now. Like AlexNet, the network believes that the increase of the depth of convolution neural network and the use of small convolution kernels have a great effect on the final classification and recognition effect of the network [12]. However, it also loses the real-time performance of detection and is difficult to be used in practice.

TABLE II. RESULTS

BaseModel	***params***	***RMSE(m=500)***	***FPS***
VGG16(ReLU) MobileNetV1(ReLU) MobileNetV2(ReLU)	26M	0.89	46
	7M	1.25	67
	6M	0.93	85
VGG16(SeLU) MobileNetV1(SeLU) MobileNetV2(SeLU)	26M	0.82	47
	7M	1.19	67
	6M	0.87	85

The VGG16 + SSD and MobileNet + SSD detection frameworks are constructed, in which the input size of the picture is 300 * 300.The parameters of the model include the number of bbox, scales, aspect_ratios , position offset, and so on. The weight of the basic feature network is selected by ImageNet pre-training weights, and the number of training rounds is 10000+. Finally, a video is selected for comparison. Finally, a video is selected for comparison, and the RMSE is calculated based on the actual number of people and the number of people in each frame. A series of statistics were made on the operating parameters and efficiency of the model. Table 2 compares the results and compares them in terms of parameter size, model accuracy, and model efficiency.

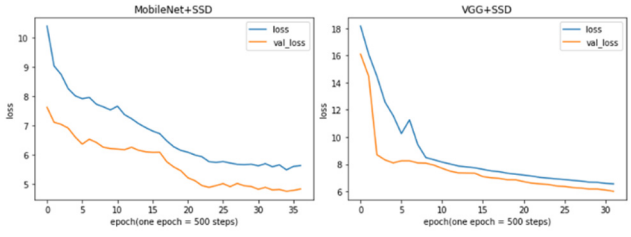

Fig.5 In Loss, the contrast between MobileNet and VGG

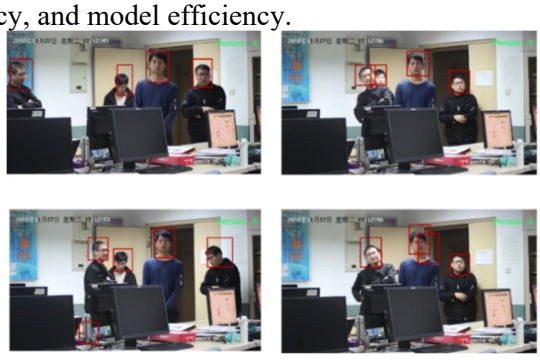

Fig.6 Partial visualization results for MobileNetV1+SSD

Analysis of the results: 1) Through the visualization of the loss (Fig 7), it can be seen that both MobileNet and VGG can converge to a range more quickly, showing only 0-40epoch results (one period = 500 steps); 2) all models are Verify on the video (Fig 8). The experiment saves each frame of

the video as an image and extracts some of the results from this article to the network MobileNet + SSD. There are several types of phenomena: a) better recognition persons with bow and partial occlusion; b) cases with recognition errors; c) cases with repeated face detection 3) Comparing the final evaluation criteria RMSE (Table II), MobileNet is slightly worse than VGG. In real-time testing, the time to infer a frame is much better than VGG. In practical applications, it usually involves modules such as video capture, encoding, preprocessing and decoding. This is a work that needs continuous improvement, and it is hoped that it will be real-time in the future.

4. Conclusion & Future

In this paper, the existing statistical algorithms in various fields are analyzed. Starting from the two directions of efficiency and accuracy, this paper proposes a population statistics algorithm based on lightweight network MobileNet, and makes some optimization to improve the robustness of the algorithm. The labeling of the data set is modified for the scenario of population statistics. Three kinds of data sets are used for experiments, and data enhancement is added to the later training. Compared with other complex networks, it loses a bit of accuracy, but in real time, it has been greatly improved.

It is a good direction to do some work on the number of people based on image processing: one is that in the selection of the data set, the characteristics of the crowd in different scenes have a certain tendency, such as the crowd on the bus, the body has occlusion, but The head information is relatively complete; the crowd at the mouth of the passage, the different passages, the density of the crowd and the way of moving are different. The characteristics of the single passage are very comprehensive, which is very suitable for counting statistics and identifying relevant. The second is to do more research work on model compression, such as model structure, model parameter precision transformation, distributed parallel computing and so on. At the same time, for the statistical application of people in different scenarios, different model structures are proposed, or a certain prior knowledge is extracted and model training is added. The algorithm model can automatically adapt to the statistics of people in different scenarios, which is a research direction in the future.

References

[1] Howard A G , Zhu M , Chen B , et al. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications[J]. 2017
[2] Liu W , Anguelov D , Erhan D , et al. SSD: Single Shot MultiBox Detector[J]. 2015.
[3] Klambauer G, Unterthiner T, Mayr A, et al. Self-Normalizing Neural Networks[J]. 2017.
[4] Zhang Junjun, Shi Zhiguang, Li Jicheng. Research Status and Trends of Population Statistics and Population Density Estimation Techniques [J]. Computer Engineering and Science, 2018.
[5] Hashemzadeh M , Farajzadeh N . Combining Keypoint-Based and Segment-Based Features for Counting People in Crowded Scenes[M]. Elsevier Science Inc. 2016.
[6] Qinglong C, Hongshan X, Li Ning. A statistical method for population size based on normalized foreground and corner information[J]. Journal of Electronics & Information Technology, 2014(2).
[7] Dalal N , Triggs B . Histograms of Oriented Gradients for Human Detection[C]// null. IEEE Computer Society, 2005.
[8] Viola P , Jones M . Rapid Object Detection using a Boosted Cascade of Simple Features[C]// null. IEEE Computer Society, 2001.
[9] Gao Fei, Minqiang F, Minqian W, et al. Research on the Method of Population Statistics Based on the Definition of Hot Spots[J]. Computer Science, 2017(S1): 183-188+211.
[10] Guoshu Z, Qiuzhen Z, Wang Hui. Statistics of video indoors based on deep learning SSD model[J]. Industrial Control Computer, 2017(11): 51-53.
[11] Sandler M, Howard A, Zhu M, et al. MobileNetV2: Inverted Residuals and Linear Bottlenecks[J]. 2018.
[12] Simonyan K , Zisserman A . Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. Computer Science, 2014.

Prosody Prediction with Discriminative Representation Method

Jipeng Zhang †
School of Information Science and Engineering Xinjiang University of China; Xinjiang Key Laboratory of Signal Detection and Processing
zhangjipeng@stu.xju.edu.cn

Hankiz Yilahun*
School of Information Science and Engineering, Xinjiang University of China; Xinjiang Key Laboratory of Multilingual Information Technology, China
hansumuruh@xju.edu.cn

Xiaoqin Feng
Mobvoi AI Lab
Suzhou, China
xiaoqin.feng@mobvoi.com

Yunlin Chen
Mobvoi AI Lab
Suzhou, China
yunlinchen@mobvoi.com

Xipeng Yang
Mobvoi AI Lab
Suzhou, China
xipeng.yang@mobvoi.com

Askar Hamdulla
School of Information Science and Engineering
Xinjiang University
Xinjiang Key Laboratory of Multilingual Information Technology, China
askar@xju.edu.cn

***Abstract*—Rhythm affects the naturalness and intelligibility of Text-To-Speech (TTS). However, rhythm prediction remains a great challenge, usually in two aspects: 1) the united annotation is a relatively difficult task, which depends on expert's experience. 2) traditional methods based on conditional random field (CRF), which heavily rely on feature engineering, such as word segmentation, part of speech(pos) etc. For above problems, we propose a method to reduce the dependency for united annotation data and conduct the joint experiment which use one unified model on independent data. Meanwhile, we also propose an algorithm of Layer Look Up Table (LLUT): use an embedding layer to learn a discriminative representation for different level of prosody data without any feature engineering. By using this method, the classifier can share the parameters and predict for different prosody level separately, which reduces the number of trainable model parameters. In order to better represent the input text, we use the pre-training model, like BERT, to provide the semantic information. Our experiment shows that the method of LLUT, is better able to acquire the discriminative meaning of different prosody levels. And also, our algorithm is proved to be general for sequence annotation tasks thus we can do extra task, like polyphone-prosody prediction.**

Keywords—prosody, CRF, layer look up table, BERT, pre-training model, discriminative representation

I. INTRODUCTION

End-to-end approaches based on deep learning have been very successful in text-to-speech (TTS) synthesis. In particular, TTS systems based on sequence-to-sequence models (e.g., Tacotron [1]) enable models to map character sequences directly to acoustic features, thus eliminating the need for complex text processing front-ends. The front-end of a TTS system depends on the language. For Chinese, it includes various modules, such as text normalization, segmentation, G2P conversion [2], and prosody prediction [3][4]. In this study, we focus on Chinese prosody prediction, which is one of the important tasks to improve naturalness of speech. Previous studies have utilized traditional statistical methods [5][6] such as decision trees, HMM, CRF, etc., and several experiments have shown that CRF works best in the task of prosody prediction.

However, in Chinese speech synthesis, CRF-based rhythm prediction has two main shortcomings. Firstly, it relies on strictly united annotation data, which requires high quality for data annotation. second, it relies on strict feature engineering, which requires high experience of annotators; finally, it cannot fully exploit the semantic information of the text by constructing statistical windows of the context.

In recent years, the research on text representation has entered the stage of state-of-art, among which research led by pre-trained language models enables researchers from different institutions to obtain excellent experimental results. For example, BERT [7], GPT, ELMO based on Transformer bi-directional encoder can be fine-tuned on a variety of text tasks such as QA, DG, sequence tasks, etc. to take full advantage of the pre-trained models [8]. Based on such architectures, the main contributions of this paper are as follows: (1) The conventional way of jointly labeling data requires strict labeling rules. By disentangling the different hierarchical rules of rhyming data, we solve the problem of not easily obtaining the complete joint dataset and improve the efficiency of data labeling. (2) We propose a discriminative feature representation under different prosody levels to facilitate the design and extension of multi-layer rhythm models. (3) Leverage the pre-trained model BERT to obtain rich semantic features without any feature engineering. (4) We propose several types of training methods for the discriminative representation task, which can achieve consistent results compared to training on joint data. We use the CRF model as the benchmark model to compare the proposed method in an objective evaluation based on F1 score, and meanwhile, conduct some ablation experiments to demonstrate the effectiveness of the proposed method and strategy.

†Work done during internship at Mobvoi AI Lab

II. PROPOSED METHOD

The goal of prosody prediction is to predict the correct pauses of phrases from the input text. This work can be viewed as a sequence labeling task [9], that is, after the text processing module, each character is marked with pauses or not. That is, given a source sequence $X = \{x1, x2, .., xt\}$ and the tag sequence $Y = \{y1, y2, ... yt\}$, the goal is to predict the label sequence from the character sequence. The ith element in the label sequence Y can be defined as a four-category label with non-prosodic pauses, prosodic words, prosodic phrases, or intonation phrases.

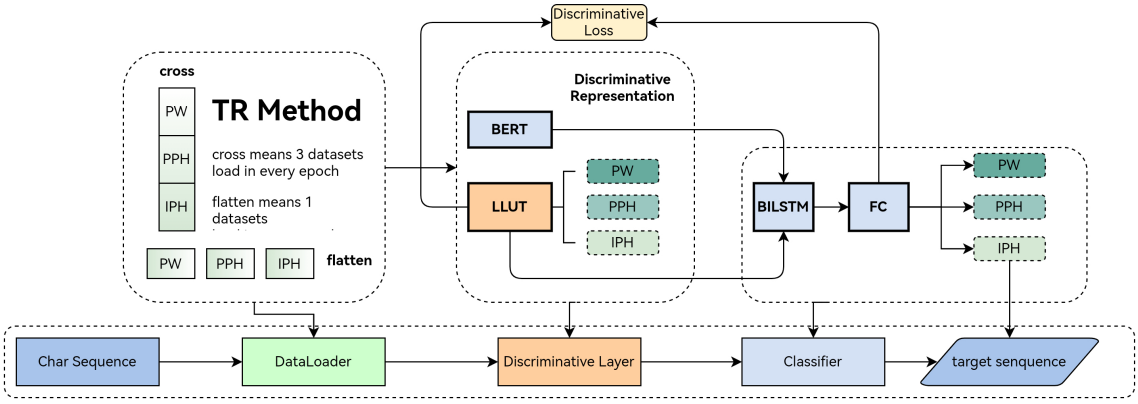

Fig. 1. Proposed model architecture: prosody prediction with discriminative representation structure

We use transformer-based BERT to model the sentence representation. The following is our proposed network structure, which mainly includes three parts: Data Loader, Discriminative Layer and Classifier. We will introduce the main structures in the following sessions, in which section A introduces the framework of BERT, section B introduces the Discriminative representation method proposed in this article, and section C introduces different training methods proposed in this article.

A. Bert Architecture

The pre-trained language model has the following advantages. Using massive unlabeled corpus data, it can learn common language representation and improve the effect of downstream tasks; it can better initialize other models, speed up model training and improve the effect; pre-training can reduce downstream tasks. Risk of overfitting on small data, equivalent to a regularization method.

BERT is a recently popular language model [10] that consists of a bidirectional Transformer encoder. The model can be used as an encoder, taking a series of characters as input and generating a word embedding for each token. The multi-headed self-attentive mechanism in the Transformer blocks enables the model to capture word dependencies in left and right-side contexts without any restrictions on word position in the sentence. After two unsupervised pre-training tasks, namely Masked Language Model (MLM) and Next Sentence Prediction (NSP), BERT is used to fine-tune the downstream tasks. Thanks to this pre-training, the model is believed to be able to capture rich Chinese contextual and semantic information, thus facilitating subsequent NLP tasks. In this paper, namely, we use BERT as sentence representation for prosody prediction, and introduce lstm to further extract the pre-trained features by adding sequence loss crf or softmax to construct the loss between target and prediction.

B. Discriminative Representation Structure

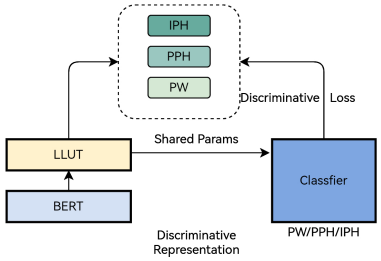

Fig. 2. Discriminative representation structure

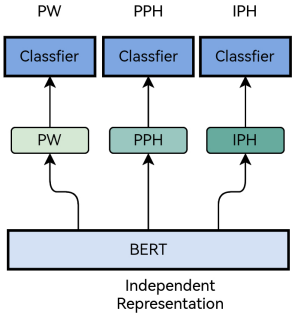

Fig. 3. Independent representation structure

After general modeling of sentences based on Bert, we constructed a discriminative representation called LLUT (Layer Look-Up Table). The motivation of this approach is to model the different level of prosody, achieving the effect of both differentiating the representation [11] classes and sharing the parameters of the intermediate layer as well as the classification layer for joint learning. We import an embedding layer to construct this module. During training, the corresponding layer is passed in and the embedding of the layer is contact to the output layer of Bert, where this embedding can be trained and updated to continuously adjust to obtain the best differentiated representation. In addition, our losses are also calculated for the current layer type and are not calculated jointly. We believe that this is a discriminative loss scheme, and learns the connection between layers in the intermediate shared parameters. We later performed a series of ablation experiments to demonstrate the effectiveness of our proposed method. At the same time, our proposed method is a general methodology, which will play a certain guiding role for similar tasks.

In this paper, softmax based classification loss as well as CRF based sequence loss are chosen for the experiments, where the corresponding objective functions are as follows:

a) Softmax based classification loss

Using cross-entropy as the training loss, the loss of rhythm labels is calculated and then averaged over the training sentences as follows:

$$L = -\frac{1}{|W^x|}\sum_{\omega \in W^x}\sum_{c} 1\{c=k_\omega\} \times \log y_c \quad (1)$$

where w^x is the index set of rhyming words in the training data, $\mathbb{1}$ is the indicator function, and k_ω is the true label of character ω.

b) CRF based sequence loss

During training, the model is optimized by maximizing the fraction of correctly labeled sequences = $(y1, y2, .., yT)$ while minimizing the fraction of all other sequences.

$$E = -s(y) + \log \sum_{\bar{y} \in \bar{Y}} e^{s(\bar{y})} \quad (2)$$

where s(y) is the CRF score of the sequence, and $s(\bar{y})$ denotes all possible sequences of labels.

C. A Special Training Method

Comparing to the normal rhythm task, we also inspired different training methods because of the inconsistency in the way the data are constructed. For the disentangling prosody dataset, we propose two training methods, which are mainly distinguished by the input method of the data in the training phase and the setting of the corresponding BERT parameters. Among them, method 1-Flatten, as shown in the left of Figure 2, uses different Data Loaders for PW, PPH and IPH, respectively. In each epoch, only one type of data will be sent to the model, that is, the training will be performed in the order of prosody level. The point to note in this method is that the parameter update strategy of the pre-training layer will be adjusted accordingly with the design of the model to suit different model structures. Method 2-Cross, use one Data Loader for PW, PPH, IPH data. At each epoch the three types of data are fed into the model according to the crossover, that is, the crossover of rhyme data is trained. For this reason, different model structures will be able to fine-tune the parameters of the pre-trained parameter layers. We believe that the cross method should be better than flatten, because it can optimize the parameters of the model by cross, rather than the way of sequential coverage. We argue that it is the Discriminative representation that does the trick, completely distinguishing the parameter space of each layer, as we will verify and explain in the experimental section.

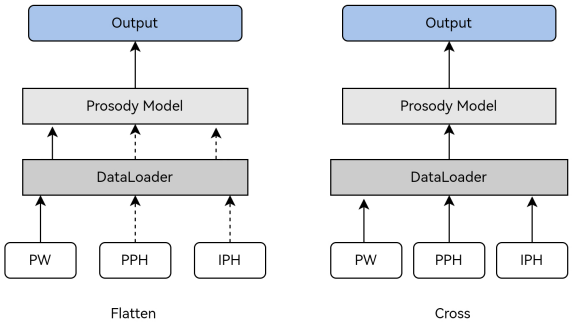

Fig. 4. Different training method design

III. EXPERIMENTS

A. Experiment Setting

Since there is no public dataset for the Chinese prosodic boundary prediction task, the experimental data used in this paper comes from within the company and is annotated by experienced language experts. The annotation results were reviewed to ensure consistency and accuracy. The annotation results are reviewed to ensure consistency and accuracy. Each rhyme in the dataset contains approximately 130,000 sentences, divided into a training set, a validation set, and a test set in an 8:1:1 ratio. For all experiments, we use a 2-layer BiLSTM (with 512 dimensions). The model is trained for 20 rounds using Adam as the optimizer, with batch size and learning rate set to 64, e-5, respectively. The loss functions we use to train the model are cross-entropy loss and CRF loss, while the model is evaluated with F1 scores. If the F1 value does not increase after 20 rounds, we stop the training early.

B. Contrast Experiment I

This set of experiments is mainly to verify the validity of our proposed discriminative representation structure and the experimental comparison with CRF as the benchmark model. Bert-Independent means that three independent classification layers are used for each of the three rhythm levels, and the parameters are updated independently between the layers; Bert-Cascade means that on the basis of Bert-Independent, the input of different levels takes into account the output of the previous level, which belongs to the normal cascade data training method. Bert-LLUT is the proposed model in this paper, in which different levels are represented in the representation layer and share the classification layer, which can be described in Section II. We choose the training method of cross, and the experimental results are shown in Table II.

TABLE I EXPERIMENTAL RESULTS OF BENCHMARK MODELS AND PROPOSED MODEL

Model	Label	ACC	REC	F1
CRF	PW	0.948	0.937	0.927
	PPH	0.908	0.927	0.745
	IPH	0.960	0.646	0.733
BERT-LLUT	PW	0.965	0.963	**0.964**
	PPH	0.922	0.924	**0.923**
	IPH	0.839	0.801	**0.822**

The experimental results in Table I show that our model shows a great improvement in the metrics of ACC, REC and F1 values compared with the baseline experiment.

TABLE II EXPERIMENTAL RESULTS OF THREE DIFFERENT MODELS

Model	Training Method	Loss	PW	PPH	IPH
Bert-Independent	cross	softmax	0.961	0.921	**0.831**
Bert-Cascade	cross	softmax	0.961	0.920	0.820
Bert-LLUT	cross	softmax	0.962	0.920	0.818
Bert-Independent	cross	crf	0.962	0.921	**0.823**
Bert-Cascade	cross	crf	0.962	0.923	0.827
Bert-LLUT	cross	crf	0.962	0.921	0.827

The experimental results in Table II show that our proposed representation of Discriminative is able to achieve parity with the normal training method for the rhythm classification task, demonstrating the effectiveness of the method.

C. Contrast Experiment II

This set of experiments is mainly designed to verify the effectiveness of our proposed training method. Two types of training methods, Flatten and Cross, are included, and the specific method theory is introduced in Section II. We selected three different types of model structures, and it is known from Table II that crf is effective, so we selected crf as the loss function and used F1 values of different rhythm levels for evaluation, and the experimental results are shown in Table III.

TABLE III EXPERIMENTAL RESULTS OF DIFFERENT TRAINING METHODS

Training Method	Model	PW	PPH	IPH
cross	Bert-Independent	0.962	0.921	**0.831**
flatten	Bert-Independent	0.956	0.904	0.793
cross	Bert-Cascade	0.961	0.920	**0.827**
flatten	Bert-Cascade	0.956	0.906	0.794
cross	Bert-LLUT	0.962	0.920	**0.828**
flatten	Bert-LLUT	0.948	0.879	0.782

With the above result analysis, the cross-based training approach is superior to the flatten approach, which is consistent with our expected results. There is roughly a 1-3 percentage point improvement in the results of each layer. We believe that it is the discriminative representation that does the trick, fully differentiating the parameter space of each layer for learning. Further, we also see that the cascade and flatten model architectures cause little impact on the experimental results.

IV. CONCLUSION

In this paper, inspired by the great success of BERT in many NLP tasks, we propose a prosody prediction model based on the LLUT approach. Through the exploration of the model structure, the experiment proves the effectiveness of our proposed method. 1)By disentangling the different hierarchical rules of prosodic data, we solve the problem of obtaining united annotation data 2)Under the non-joint prosody dataset, a discriminative representation method is proposed towards different prosody levels, which effectively learns the discriminative representation 3)Use the pre-trained model BERT to obtain rich semantic features without any feature engineering 4)Several training methods for the discriminative representation task are proposed, which achieve a better result.

However, we still have many shortcomings. For example, we can propose better methods on data annotation for better rhythm consistency. And also, we will explore the following aspects in the future 1) choosing more pre-trained language models for experiment 2) exploring better data representation methods 3) following lightweight deployment of models.

ACKNOWLEDGEMENT

We would like to thank Mobvoi TTS labeling team in Wuhan for supporting data processing and labeling. Thanks also to our colleagues Yunlin Chen, Xiaoqin Feng and Xipeng Yang et al. for helpful discussions and advice. This work was supported by the Strengthening Plan of National Defense Science and Technology Foundation of China (2021-JCJQ-JJ-0059) and Natural Science Foundation of China (U2003207).

REFERENCES

[1] Wang Y, Skerry-Ryan R J, Stanton D, et al. Tacotron: Towards end-to-end speech synthesis[J]. arXiv preprint arXiv:1703.10135, 2017.

[2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[3] Xie K, Pan W. Mandarin prosody prediction based on attention mechanism and multi-model ensemble[C]//International Conference on Intelligent Computing. Springer, Cham, 2018: 491-502.

[4] Futamata K, Park B, Yamamoto R, et al. Phrase break prediction with bidirectional encoder representations in Japanese text-to-speech synthesis[J]. arXiv preprint arXiv:2104.12395, 2021.

[5] Qian Y, Wu Z, Ma X, et al. Automatic prosody prediction and detection with Conditional Random Field (CRF) models[C]//2010 7th International Symposium on Chinese Spoken Language Processing. IEEE, 2010: 135-138.

[6] Zheng Y, Tao J, Wen Z, et al. BLSTM-CRF Based End-to-End Prosodic Boundary Prediction with Context Sensitive Embeddings in a Text-to-Speech Front-End[C]//Interspeech. 2018: 47-51.

[7] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

[8] Radford A, Narasimhan K, Salimans T, et al. Improving language understanding by generative pre-training[J]. 2018.

[9] Zheng Y, Tao J, Wen Z, et al. BLSTM-CRF Based End-to-End Prosodic Boundary Prediction with Context Sensitive Embeddings in a Text-to-Speech Front-End[C]//Interspeech. 2018: 47-51.

[10] Qiu, Xipeng, et al. "Pre-trained models for natural language processing: A survey." *Science China Technological Sciences* 63.10 (2020): 1872-1897.

[11] Che H, Li Y, Tao J, et al. Investigating effect of rich syntactic features on mandarin prosodic boundaries prediction[J]. Journal of Signal Processing Systems, 2016, 82(2): 263-271.

A Method of Quick Edge Detection Based on Zynq

Rong Xie
Department of information, Beijing University of Technology
Beijing, China
xierong8989@163.com

Xiaoqin Feng
Department of information, Beijing University of Technology
Beijing, China
fengxqinx@163.com

***Abstract*—Most of existing image processing algorithms have poor real-time performance in embedded device applications and occupy too much software resources. In this paper, a fast edge detection method is proposed based on Zynq platform. The xfOpenCv acceleration library provided by Xilinx is used to implement the edge detection algorithm. Specifically, the function operation process is accelerated in the FPGA through the acceleration constraint of a specific function implemented in the xfOpenCv library. In the implementation process, the SDx development environment is also used to simplify the function hardware acceleration development process. The experimental results show that our edge detection method takes less time than other methods, and has better real-time performance with the basically same detection results.**

Keywords—edge detection, Zynq, hardware acceleration

I. INTRODUCTION

Image processing technology originated in 1920s. Early image processing is mainly based on human objects to improve the quality of images and meet the visual effects of people. In the 1960s, with the development of electronic computer technology, people began to use computers to process image information. With the deepening of research, people began to study how to use computers to analyze and interpret images [1]. Its process resembles human visual understanding of the external world.

With the rapid development of electronic technology, more and more traditional digital image processing system can be realized through embedded system. Embedded system has a series of advantages such as small size, low cost, low power consumption and good reliability. The embedded image processing system based on embedded technology has been widely used in industrial testing, machine vision, aerospace, military guidance, biomedicine, public safety, Car driving auxiliary and other fields [2]. At present, embedded processors used in image processing mainly include ARM, DSP and FPGA. However, the existing embedded image processing solutions based on a single ARM processor, DSP or FPGA can no longer meet the requirements of high-performance applications. Using multiple chips combined solution will often result in complex hardware architecture, difficult development, system instability and other issues. In response to the above issues, the paper uses Xilinx Zynq all-programmable platform. The platform is the first heterogeneous chip that tightly integrates the high-performance ARM Cortex A9 hardcore and the programmable logic FPGA. Through this combination, Zynq not only has the transaction management, operating system and other advantages in ARM processor, but also has parallel processing, the dynamic reconfiguration and other advantages in FPGA [3, 4, 5].

This paper designs a fast edge detection method under the Zynq platform using the xfOpenCv kernel proposed by Xilinx. The xfOpenCv is a library designed in Xilinx SDx development environment. xfOpenCv is an improvement on OpenCv made by Xillinx for embedded environments. Its functions are mostly similar in functionality to their OpenCV equivalent. It's intended for application developers using Zynq-7000 All Programmable SoC and Zynq UltraScale+ MPSoC devices. It provides a software interface for computer vision functions accelerated on an FPGA.

II. THE METHOD OF EDGE DETECT

A. Canny edge detection algorithm

This paper uses canny edge detection algorithm as an example. Canny edge detection can be divided into four steps: image smoothing, gradient and direction calculations, non-maximum suppression, detect and connect edges. The first step: image smoothing. The Canny algorithm uses a two-dimensional Gaussian function to smooth the image, shown in Equation (1).

$$G(x,y)=\frac{1}{2\pi\ \sigma^2}\exp(-\frac{x^2+y^2}{2\sigma^2}) \qquad (1)$$

In Equation (1), σ is a Gaussian filter parameter, which controls the degree of smoothness. The filter with smaller σ has high accuracy of positioning but low signal to noise ratio. The filter with large σ is just the opposite. Therefore, we need to select Gaussian filter parameters as required. The second step: gradient and direction calculations. The Canny algorithm uses a first-order differential operator to calculate the gradient magnitude and gradient direction at each point of the smoothed image to obtain the corresponding gradient magnitude image G and gradient direction image θ. The partial derivatives at points (i,j) in both directions are $G_x(i,j)$ and $G_y(i,j)$, shown in Equation (2) and (3).

$$G_x(i,j)=(I(i,j+1)-I(i,j)+I(i+1,j+1)-I(i+1,j))/2 \qquad (2)$$

$$G_y(i,j) = (I(i,j) - I(i+1,j) + I(i,j+1) - I(i+1,j+1))/2 \quad (3)$$

At this point, the gradient magnitude and gradient direction at point (i,j) are $G(i,j)$ and $\theta(i,j)$, shown in Equation (4) and (5).

$$G(i,j) = \sqrt{G_x^2(i,j) + G_y^2(i,j)} \quad (4)$$

$$\theta(i,j) = \arctan\left(\frac{G_x(i,j)}{G_y(i,j)}\right) \quad (5)$$

The third step: non-maximum suppression. In order to accurately position the edges, the ridge band in the gradient-magnitude image G must be refined to only retain the local maxima of the amplitude, ie non-maximal suppression (NMS). The Canny algorithm interpolates along the gradient direction $\theta(i,j)$ in the neighborhood of 3 × 3 with the point (i,j) as the center in the gradient image G. If the gradient value $G(i,j)$ at the point (i,j) is larger than the two adjacent interpolation values in the direction of $\theta(i,j)$, the point (i,j) is marked as a candidate edge point. Otherwise, it is marked as a non-edge point. The fourth step: detect and connect edges. The Canny algorithm uses double-threshold method to detect and connect the final edge from candidate edge points. The double threshold method first selects the high threshold T_h and the low threshold T_l, and then starts scanning the image. It is detected any pixel point (i,j) marked as a candidate edge point in the candidate edge image. If the point (i,j) gradient magnitude $G(i,j)$ is higher than the high threshold T_h, the point is considered to be an edge point. If the point (i,j) gradient magnitude $G(i,j)$ is lower than the high threshold T_l, the point is considered not to be an edge point. For pixel points whose gradient amplitude is between the two thresholds, they are regarded as suspected edge points and need to be further judged according to the connectivity of the edges. If there are edge points in the adjacent pixels of the pixel point, the point is also considered to be an edge point. Otherwise, it is considered to be a non-edge point [6, 7, 8]. Through the four-step processing of the picture by the canny algorithm, the edge of the picture can be well detected. On the other hand, using the Canny operator edge detection algorithm, the detection result image can also respond better to the weaker edges.

B. The advantages of edge detection base Zynq

In the field of embedded image processing, due to the constraints of system architecture, memory bandwidth and other factors, it is difficult for traditional single-chip microcomputer systems to meet the requirements of throughput and computer performance at the same time. In 2011, Xilinx introduced the Zynq-7000 series of scalable platforms that integrate programmable logic with the dual-core ARM A9 processing system. With integrated ARM processors, Zynq-7000 leverages existing embedded resources. The integrated FPGA resources allow the Zynq-7000 to provide high-performance computing solutions [9]. Based on the significant advantages of the ZYNQ platform, the video image processing system under the embedded platform mainly includes a programmable logic (PL) and a processing system (PS). Part of the PL is mainly responsible for the control logic of each module, including the DMA transmission of video image data and HD display. It can achieve hardware acceleration of video image processing algorithms in the high-level synthesis tools. PS part is mainly responsible for running Linux system on ARM Cortex A9 [10].

In the general Zynq development process, the OpenCv library needs to be ported to the Linux operating system running in the arm kernel in Zynq. Running image processing programs in Zynq need to cross-compile and call the ported OpenCv library file to execute the program. In practical applications, there is a problem that image processing methods based on the ARM platform are too slow in running complex algorithms [11, 12]. XfOpenCv library is Xilinx optimizes OpenCv with FPGA. XfOpenCv can be used in the device application development of Zynq-7000 All Programmable SoC and Zynq UltraScale+ MPSoC. xfOpenCV library has been designed to work in the SDx development environment, and provides a software interface for computer vision functions accelerated on an FPGA device. Figure 1 shows the architecture of xfOpenCv in Zynq.

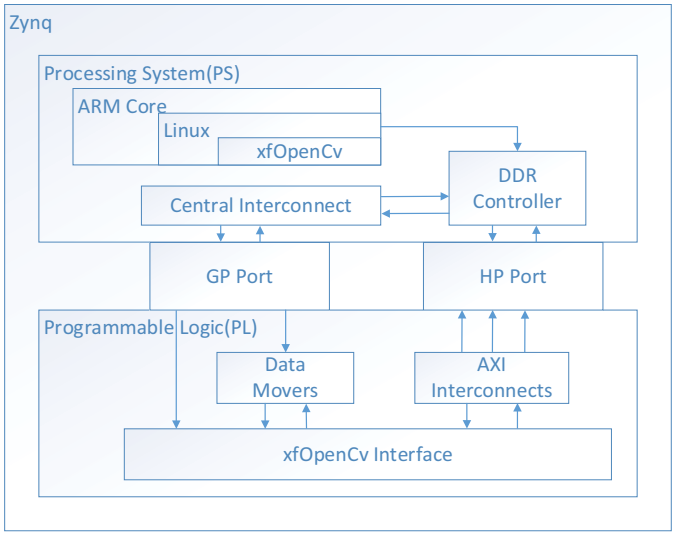

Figure. 1 The architecture of xfOpenCv in Zynq

III. THE DESIGN OF A FAST EDGE DETECTION METHOD

This paper implements a fast edge detection method by Xilinx xfOpenCv libraries. The implementation process applies to the high-level synthesis tools Vivado HLS and Xilinx Zynq SOC all-programmable processing platform, Vivado image video library, and converts OpenCv to RTL through the advanced synthesis tool Vivado HLS. Hardware acceleration of

OpenCv program algorithms is implemented on Xilinx Zynq SoC series all-programmable processing platforms. Take the Canny edge detection as an example.

A. The implementation of edge detection method

This paper takes the Canny algorithm as examples. It is based on Xilinx xfOpenCv library design and implementation. In this method, the Mat data format needs to be converted to the hls::stream data format. The edge detection process of the Canny algorithm is divided into four steps: image filtering, gradient calculation, non-maximal suppression, and connection edges. After processing, it convert the hls::stream data type to the Mat data format. The processing flow of Canny algorithm figure shown in Figure 2. The data format of the input image needs to be converted. The data format of the picture read from by OpenCv is Mat data type. The Mat data type is converted into xf::Mat type data through the copy operation. The xf::Mat data format is an intermediate format that prepares for the next data format conversion to hls::stream. The data of xf::Mat type needs to assign each row and column data to the data of hls::stream type. The assignment process uses two loop processes: row loop and column loop. The two loop processes are optimized using the existing optimization method in HLS. There are two optimizations for the loop: the pipeline constraint and the dataflow constraint. The pipeline constraint causes the loop in the function to be pipelined. Pipelined execution loops increase the parallelism of operations. Because there is a data transfer relationship between the row loop and the column loop, the dataflow constraint is added between the two loops so that different loop bodies execute in parallel and the throughput of the data can be improved. Converting Mat data to the hls::steam data format allows the program to achieve optimal performance during execution. After the data format conversion is complete, the canny algorithm of edge detection starts the edge detection part of the calculation. The first step is the Gaussian filtering process. In the Gaussian filtering process, we need to matrix convolute for each pixel of the image. The matrix convolution can be divided by ARRAY_PARTITION constraint. In Gaussian filtering calculation, the kernel number of Gaussian convolution is three, so the matrix can be divided into three parts by using ARRAY_PARTITION constraint. The parallel bandwidth and the computing power in the calculation process of the divided matrix. On the other hand, the #pragma HLS RESOURCE variable=buf core=RAM_S2P_BRAM constraint adds the matrix buf to the RAM. It controls the delay of generating BRAM to ensure the correct timing in the matrix operation. The second step is to use the Sobel algorithm to calculate the image gradient, which is the same as the Gaussian filtering process. The gradient of the calculated graph also involves matrix convolution operations. It also uses ARRAY_PARTITION constraint to accelerate the convolution of the matrix. The third step is non-maximum suppression. The specific operation process is to perform a convolution operation with a 3*3 filter kernel and the input image. The calculation process also involves a convolution operation. The ARRAY_PARTITION constraint is also used to accelerate the matrix convolution operation speed. The fourth step is the process of connecting the edges. It uses the high and low thresholds set by the user to determine if the pixel is an edge point. The computation process optimizes the loop process through pipeline constraints, which improves parallel computing capabilities. After the Canny algorithm is completed, the image data format is converted from hls::stream to xf::mat, and finally restored to the Mat type. The entire computational acceleration process utilizes the parallelism of the FPGA. It accelerates the matrix convolution and the cyclic process through parallel operations that improves the speed of image processing and the real-time performance in real-time video processing.

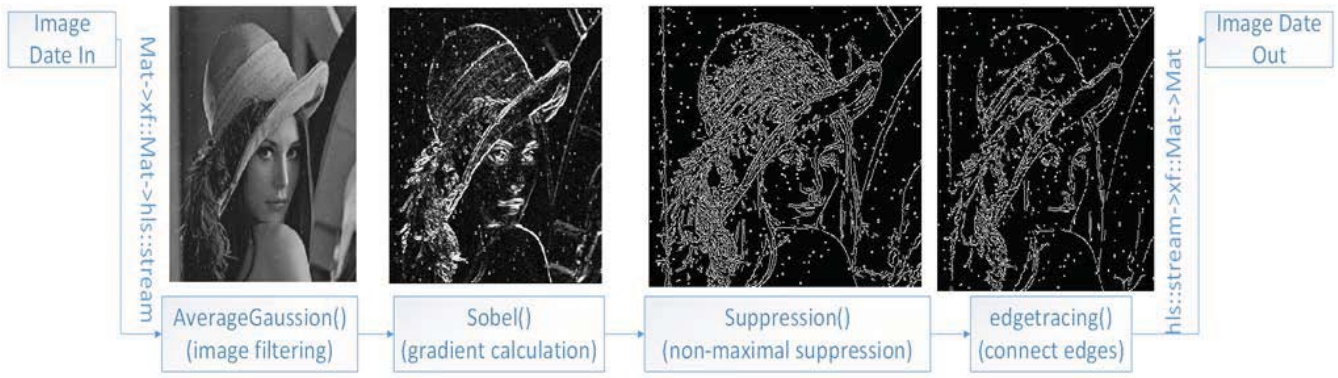

Fig. 2 The processing flow of canny algorithm

B. The contrast experiment

This paper takes edge detection of Canny algorithm as an example, and compares the time spent and the quality of the results in the three cases. The experimental environment configured on the PC is Ubuntu16.4 and OpenCv3.4.1. The development board used in the experiment is the ZUC102 development board under the Zynq series of Xilinx. The development board experimental environment is to transplant the Linux system on the PS side. Cross-compilation of OpenCv source code for the development board support library files ported to the development board, so that the development board can support cross-compiled OpenCv program. The experiment needs to establish a Zynq-based development project in SDx, and write the edge detection source code. The most critical steps is to put hardware-accelerated parts of code into hardware for acceleration. The part that can be accelerated in this experiment is the Canny algorithm part. The functions implemented in the hardware call FPGA resources at runtime to achieve parallel acceleration, improve program execution

speed, and save software resources. The results of edge detection in three environments are shown in Figure 3.

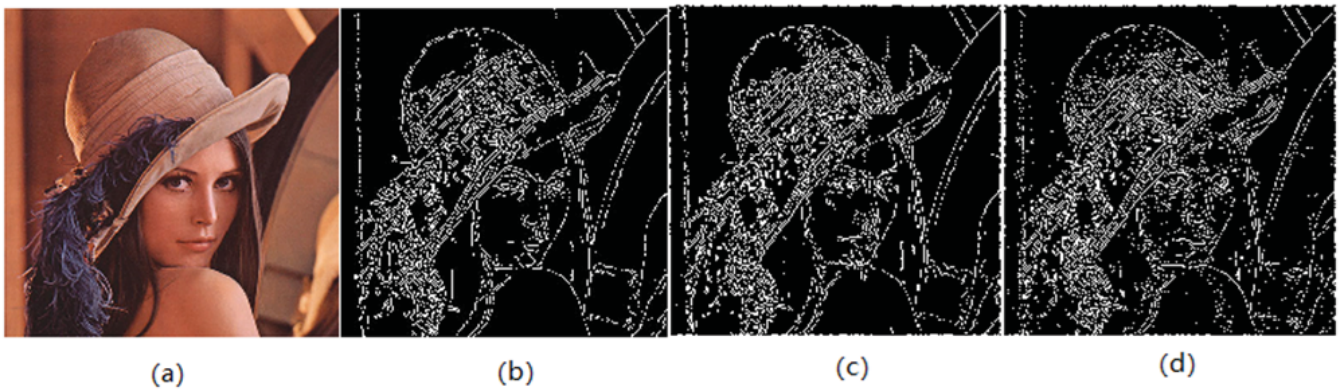

Fig. 3 The result of edge detection in three environments

Figure 3 (a) the original image input. (b) the result of PC environment. (c) the result of unaccelerated in the development board environment. (d) the result of accelerated in the development. The edge detection process in the three environments sets the same high and low thresholds, and the result of the edge detection is basically the same. Programs that are accelerated using the xfOpenCv library output the same results as the general case. Comparing program runs takes time in the same situation where edge detection results. Experiments tested the time-consuming of edge detection for twenty different pictures. Time-consuming in the three environments are shown in Table 1.

TABLE I. TIME-CONSUMING IN THE THREE ENVIRONMENTS

Enviement	**PC**	**OpenCv of board**	**Ours**
Highest time-consuming(s)	0.009230	0.072895	0.005337
Lowest time-consuming(s)	0.006940	0.060540	0.004096
Average time-consuming(s)	0.008030	0.066191	0.004170

It can be concluded that the accelerated algorithm has a significant reduction in time-consuming. The use of xfOpenCv to accelerate the canny algorithm for edge detection has some advantage in time-consuming than other environments.

IV. SUMMARY

In this paper, a method of quick edge detection is proposed based on Zynq By using the xfOpenCv library. Our method uses existing acceleration methods to accelerate the edge detection process, and utilizes the parallel computing capabilities of FPGAs to speed up the edge detection process. Compared with the operation of the edge detection algorithm in other environments, our method can obtain the detection results that are basically consistent with the general case. However, it has greatly reduced the computational time.

Further studies beyond this work include improving the accuracy of edge detect result and test the performance of other algorithm in our acceleration environment, and applying accelerated thinking to other applications.

REFERENCES

[1] Pauwels K, Tomasi M, Alonso J D, et al. A Comparison of FPGA and GPU for Real-Time Phase-Based Optical Flow, Stereo, and Local Image Features. IEEE Transactions on Computers. Vol. 61 (2012) No. 7, p. 999-1012.

[2] Farabet C, Martini B, Akselrod P, et al. Hardware accelerated convolutional neural networks for synthetic vision systems IEEE International Symposium on Circuits and Systems. IEEE, 2010, p, 257-260.

[3] Appiah K, Hunter A, Dickinson P, et al. Accelerated hardware video object segmentation: From foreground detection to connected components labelling. Computer Vision & Image Understanding. Vol. 114 (2010) No. 11, p. 1282-1291.

[4] Honegger D, Oleynikova H, Pollefeys M. Real-time and low latency embedded computer vision hardware based on a combination of FPGA and mobile CPU. Ieee/rsj International Conference on Intelligent Robots and Systems. IEEE, 2014, p. 4930-4935.

[5] Savarimuthu, Rajeeth T, Kj, et al. Real-time medical video processing, enabled by hardware accelerated correlations. Journal of Real-Time Image Processing. Vol. 6 (2011) No. 3, p. 187-197.

[6] Wei-Bo Y, Du Z S. An improved Kirsch human face image edge-detection method based on canny lgorithm. International Conference on Consumer Electronics, Communications and Networks. IEEE, 2011, p. 4740-4743.

[7] Pan Q S, Zhang Y, Yang Z M, et al.Design and Implementation of Image Corner and Edge Detection System Based on Zynq. Computer Science. 2017.

[8] Xin G, Ke C, Hu X,. et al. An improved Canny edge detection algorithm for color image. IEEE International Conference on Industrial Informatics. IEEE, 2012, p. 113-117.

[9] Ahmad A M, Lukowicz P, Cheng J. FPGA based hardware acceleration of sensor matrix. ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct. ACM, 2016, p. 793-802.

[10] Dang V, Skadron K. Acceleration of Frequent Itemset Mining on FPGA using SDAccel and Vivado HLS. IEEE, International Conference on Application-Specific Systems, Architectures and Processors. IEEE, 2017, p. 195-200.

[11] Russell M, Fischaber S. OpenCV based road sign recognition on Zynq. IEEE International Conference on Industrial Informatics. IEEE, 2013, p. 596-601.

[12] Cortes A, Velez I, Irizar A. High level synthesis using Vivado HLS for Zynq SoC: Image processing case studies Design of Circuits and Integrated Systems. IEEE, p. 2017:1-6.

Research on the Internet of Things Platform for Smart and Environmental Protection

Junyang Sheng[1,3], Xiaoqin Feng[2,3]

[1] Beijing Engineering Research Center for IoT Software and Systems, Beijing University of Technology, Beijing, 100124, China
[2] Xiaoqin Feng Beijing Engineering Research Center for IoT Software and Systems, Beijing University of Technology, Beijing, 100124, China;
[3] Beijing Advanced Innovation Center for Future Internet Technology, Beijing University of Technology, Beijing 100124, China
shengjuny@emails.bjut.edu.cn fengxqinx@163.com

Abstract: This article focuses on the research and implementation of an intelligent environmentally IoT platform and intelligently connects traditional environmental protection devices to the IoT Cloud platform.There are different types of environmental protection equipment,such as PM2.5, Pm10, water, sound and gas monitoring equipment.In this paper the environmental protection equipments connect to the IoT cloud platform through the smart gateway, and the communication mode of the network include 2G,3G,4G, Narrowband-IoT. The management platform supports intelligent access to devices, remote management, real-time monitoring and data analysis, prediction alarm. The vision of the Internet of Things (IoT) is a dynamic global network based on standard and interoperable communication protocols where physical and virtual things have identities, physical attributes, and capabilities and are seamlessly integrated into the existing internet infrastructure. This paper descripes the intelligent environment project which is based on Smart Environmental Gateway Management Platform. It's function position is to build a smart environmentally aware network of omni-directional interconnections between the environment and society and to realize the modernization and intelligence of environmental monitoring and control The experimental results show that the IoT platform is stable, scalable, high performance.

Keywords: Internet of Things;3G/4G/NB-IoT; Environmental Monitoring IoT Platform; Data management.

1 Introduction

In the era of Industry 4.0, IoT developed rapidly. All things in the world could be interconnected via the Internet, including some high-speed services (such as video services, etc.), and some low-rate services (such as meter reading services, etc.). According to incomplete statistics, low-speed services account for more than 67% of IoT services, and low-speed services do not have good cellular technologies to provide sufficient support [1][2]. The demand for intelligentization and industrial upgrading is becoming more and more urgent in various industries and NB-IoT technology has emerged as the times require.

978-1-5386-6005-8/18/$31.00 ©2018 IEEE

It is a cellular network-based communication technology with features such as wide-area coverage, massive access, and low power consumption. Low-rate NB-IoT technology will be mainly used in municipal applications that are less sensitive to low communication delays [3][4] . With the country's emphasis on the environmental protection industry in recent years,China's environmental protection industry has entered a new stage of development [5]. It is of great significance to understand, study and use the new generation of information technology [6].Due to the wide variety of monitoring equipment in the environmental protection industry, which involves sound, lignt, Gas, odor, water and other aspects of the monitoring index and the transmission protocols are diverse too. As a result the islands of information are formed between various IoT platform , and data sharing cannot be achieved [7].

Huawei's OceanConnect platform takes the lead in the NB-IoT field. Therefore, this article refers to the the ideal of SSD in OceanConnect platform and designed an IoT cloud platform which supports access of NB-IoT and 4G LTE. Without changing the functions and design of the original environmental protection equipments, different types and multifunctional environmental protection devices can be accessed on the platform to realize the modernization and intelligence of environmental monitoring.

This paper analyzes the functional requirements of the intelligent environmental cloud platform based on the B/S architecture. For the characteristics of environment protection equipment and its data transmission difference, On the one hand , We choose to adopt smart gateway to support multiply devices which function is the same as the DTU on the terminal side and on the other hand, open TCP port is used to support the access of 3GPRS and 4G LTE; The data communication protocol refers to the data transmission of pollutant on-line monitoring system adopted by the Ministry of Environmental Protection of the People's Republic of China; The platform is adopting the Netty framework based on NIO non-blocking technology package and mature Web development technology and it realizes the project based on B/S architecture which is of greate

practical significance.

2 Environmental platforms requirements

The IoT device management cloud platform is an intelligent environment-friendly IoT cloud platform system. The smart environmental protection platform is divided into four layers, which are the perception layer, network layer, application layer, and user layer in Figure1.

The application layer consists of two parts, divided into an access platform and a management platform. The access platform connects to the environmental protection equipments from the south, receives and saves the data reported by the equipments, conducts command interaction with the equipments, and connects to the management platform from the north. The management platform south connects to the access platform , connects users from north and provides open data interface to relevant enterprises and government departments. This platform is to assist users in connecting environmental protection devices of different models and manufacturers to a smart environmental management platform, providing environmental monitoring and early warning, intelligent analysis of big data, tracing the origin of illegal emission points, and inquiring about the surrounding pollution.

The main technical features of NB-IoT include wide coverage, strong links, low power consumption and low cost[8]. Here are the specific introductions.

2.1 Wide coverage

In the same frequency band, NB-IoT covers a wider area than existing wireless technologies and can be expanded to 100 times. Due to the adoption of mechanisms such as retransmission and low-order modulation, its penetration performance is better. The same applies to applications such as in-building, underground pipe networks, and well covers that require deep coverage.

2.2 Strong links

Compared with existing wireless technologies, NB-IoT can provide more access in the same base station. A sector of NB-IoT can support 100,000 NB-IoT terminal connections.

2.3 Low power consumption

The NB-IoT consumes 1/10 of the power of 2G. it's terminal module has a long standby time of up to 10 years.

2.4 Low cost

NB-IoT can reuse site infrastructure and can reuse radio frequency and antenna, so the deployment cost can be reduced. In addition, for the NB-IoT chip, low-speed, low-power, and low-bandwidth technologies can also reduce the cost of related modules. According to NB-IoT's main technical standards, it has the following key attributes: low-rate attributes, high-latency attributes, low-frequency sub-attributes, and weak mobility attributes [9].

3 Design for smart environmental platforms

3.1 Design Principles

The design of IoT platform should support variety bussiness needs, not only meet the common needs of different business but also support the individual needs of different business [10]. In addition, the system design of the platform should also follow the following principles: safety principles, practical principles, scalability principles and standard principles. The design of the platform takes into account the different access devices, different transmission data formats, and different communication protocols. Smart gateways are used to support access to different devices. Unified data transmission standards and platform management are based on grading device.

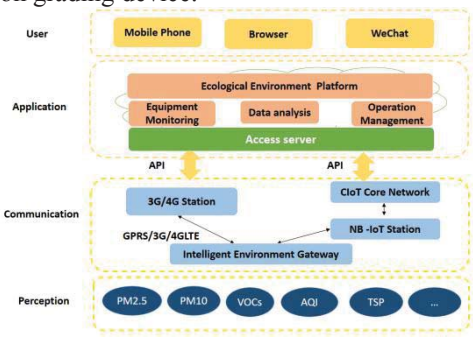

Figure 1 Ecological IoT architecture

3.2 Data format

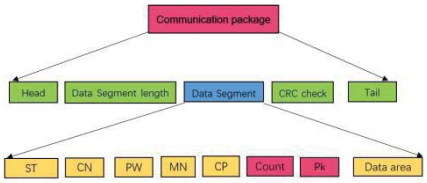

Figure 2 Communication protocol data structure

All communication packets are composed of ASCII characters (excluding Chinese S characters, using UTF-8 code,8 bits, 1 byte). The data structure of the communication protocol is shown in The communication between the device and the platform mainly includes the following four aspects: the information transmission between the user and the platform; the device actively reports the data; the platform issues the command to the device and the device responds to the platform command [11].

Terminal reporting format:

##0131ST=22;CN=2011;PW=123456;MN=781703664AM0001;CP=&&DataTime=20150811151200;PM10-Rtd=89.59,PM10-Flag=N;TSP-Rtd=133.71,TSP-Flag=N&&C241

Parameter Description:

(1) Header: Fixed as ##;

(2) Data segment length: the number of ASCII characters of the data segment, for example: length 255, then written as"0255". The data packet length is calculated from the first system number "ST=" and the data length up to the last newline character "\r\n";

(3)ST:212 System number of the agreement, "22" for air quality monitoring; 39 for site dust

(4)CN:212 The order number of the agreement,“2011” represents uploading real-time data of pollutants

(5) PW: There is no practical use, and it can be fixed as "123456";

(6)MN: It is used for device identification, a total of 15 digits, the first 9 digits are the device manufacturer's organization code, and the middle 2 digits are the device type codes (for details, see "Table 2: Device Type Codes"). Bit 4 is the unique code for this device as determined by the device manufacturer

(7) CP: (CP=&& data area &&) Please refer to the description of the data area defined in the environmental protection 212 protocol. The final format is shown in the above example.

(8) Data area: PM10 parameter is PM10; PM2.5 parameter is PM25; TSP parameter is version: V1.0 Release date: 2017.08.11 TSP; Wind direction parameter is WD; Wind speed parameter is WS; Temperature parameter is TEM; The parameter is PA; the humidity parameter is RH; the rainfall parameter is RI; the coding, measurement units, and data types are listed in the “Table 1 Gas Monitoring Factor Code Table”.

(9) Convert the CRC returned by the data from the beginning of the ST to the last two && to a hexadecimal string, added to the end of the packet. Precautions A newline character "\r\n" must be added at the end of each packet.

When the device reports other information, it also reports through this format, but the identifiers will be different. The following is the reporting of GPS coordinate information.

3.3 Platform Core Business Process

3.3.1 Add device node

The registration of devices on the platform are the core steps of the management platform. The specific registration process is as shown in the figure3.

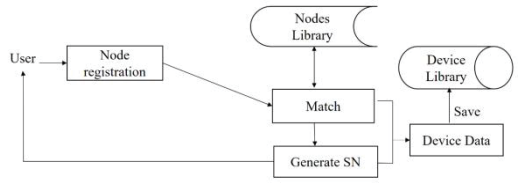

Figure 3 Add device node information to the platform

The user inputs information such as Dev_ID, Pro_ID, DevName, Manu_Name, Created_Time, and so on. When registering first, the device node will be attached to a certain project. After receiving the registration information, the operation management background will first determine whether the node is already registered. If it is registered, it will return the prompt information. If it is not registered, the user will be prompted to improve the information. After the selection is completed, the system generates an SN and saves it to the device node library.

3.3.2 Device node data adaptation

The data adapter is the process of converting the device node data into the platform's standard data format and persisting it into the database. The communication between the platform and the device can use NB or 4G mode and may also support the extension of Lora. Uploaded data is received through data interfaces such as Socket and Http. The data transmission format mainly includes JSON strings, hexadecimal strings, and so on. There are at least two different communication method in the platform, so the data analysis service needs to support them.

1) *Send via 4G format*

If data is reported by the device,the platform could determines which device by the data comes from by the BoardID, and extracts the device ID. According to the identifier of 4GLTE, the reported 4G information would be extracted, and then the network signal strength and IP address of the device would be obtained.

If the command is delivered by the platform, the server can send the command to the specified device based on the channel saved by the platform. The device can receive the message according to the command name and parameters and respond accordingly.

2) *Send data through NB*

The device Quectel_BC95 can send data to the UDP server through UDP. It only needs to provide the server port and IP.

The device Quectel_BC95 through the AT command for UDP communication.

(1) AT+NSOCR=DGRAM,17,4587,1 //Create a socket

(2)AT+NSOST=2,120.24.86.104,50000,3,AB3045

// Send data 2 is the socket number ,3 is the data length

(3) +NSONMI:0,4 // Receive data

(4) AT+NSORF=0,4 //Read data

0,192.53.100.53,5683,4,60A041C7,0 (The last 0 indicates the length of data that has not been read yet)

(5) AT+NSOCL=0 //Close a socket

3.4 Device data storage

Device data storage is divided into gateway data including temperature, GPS, and network signals. These data are stored in the database in the form of key-value pairs, and the primary key and index are used to speed up the query. In addition, the device data of GPS is stored in the FTP server . On the server, for other units to query and record.

3.5 Database Relational Model

project：

pro_id,pro_namepro_address,created_time,pro_creator,creator_phone

device：

device_id,pro_id,device_name,device_address,status,manu_name,created_time

device information：

device_id,sate_num,cur_time,longitude,latitude,board_tem,chip_tem

Environmental Data:DeviceId,PM2.5,PM10,SO2,CO,O3

User：user_name,user_pwd,user_phone,role_id

Role：role_id,role_name,role_desc,role_status

Authority：aut_id,role_id,aut_desid,aut_url,aut_des

4 Platform implementation

The platform system consists of two parts: the access server ,management platform. The function module of platform is shown in the figure4 below. There are five modules in the platform, including device monitoring, data analysis, and operations Management, server management, Statistical report.

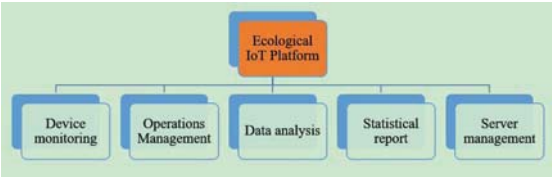

Figure 4 Function module of platform

Equipment monitoring:

Displays the engineering-equipment-gateway three-level menu. The user can perform positioning operations on the project or equipment, display the real-time geographic location of the equipment through the three-dimensional map, and the environmental status of the site.

Operation Management:

1) Effective management of the terminal equipment of the access platform:

It can display the status of the equipment in real time, including the equipment online status, temperature information, Beidou satellite quantity information, network signal strength and other status; it can add and modify the basic information of the equipment, including the equipment name, the equipment engineering, the equipment address, and equipment belonging Information such as vendor and device type; ability to delete devices

2) Ability to manage the project, including addition, deletion, modification, and inspection of the project, configuration of the sampling action of the device, remote update procedures, and remote restart of the system.

3) Ability to add, delete, modify, check, grant, and deprive users of rights to users and roles, corresponding to different projects and resources.

Data analysis:

Including monitoring data query, tracing the source, and environmental pollution warning. According to different business needs, different operations can be carried out. According to single or multiple equipment to view the environmental data, we can use large data technology and weather related algorithms to trace the source and give the corresponding evidence; the pollution level will be judged and alarm information will be issued in the future.

Statistical report and server management are support the common business of the platform.There is not much space to draw the details.

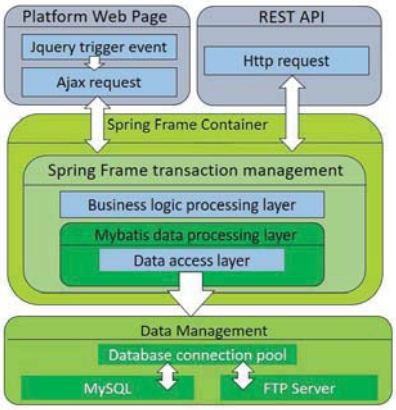

Figure 5 Technical architecture of system

The management platform and access server adopt Java Web based on the B/S architecture and use Spring+ SpringMVC+ Mybatis, The technical architecture is shown in Figure5. The access platform uses Netty4.x as the middleware to implement message communication.

For the hardware platform, the development board adopts the Zynq7010 FPGA board, the programminglanguage is the C language, and the IoT gateway device is responsible for receiving the data of the acquisition equipment and communicating with the platform and the acquisition equipment. Then report the collected data to the cloud platform through 4G or NB network. We can see the environmental monitoring equipments clearly from the intelligent environmentally IoT platform in the picture6 below. So far we have accessed a total of 30 devices.

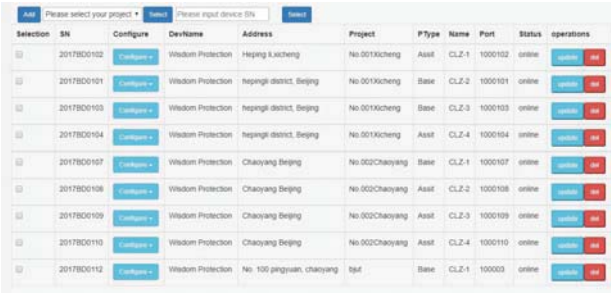

Figure 6 Device detail of platform page

5 Platform testing

The smart environmental protection equipment management platform is to achieve the access of environmental protection equipments, environmental protection equipment monitoring and data sharing. In the course of use, a large number of devices are accessed at the same time, and device communication and data query speed will affect the user experience. To understand the throughput of the platform and the response time of the data query interface, various performance tests were performed on the platform.

5.1 Data access concurrency test

Through the device access testing and concurrency testing of the platform, the results show that. The real-time performance of the device can be monitored in real-time within 15 minutes and the device is notified to the platform. When the number of accesses to the device is less than 100, the error rate is 0; the data throughput is 2725, which satisfies the concurrent requirements.

Table 1 Fest report of platform concurrent capability

Link number	Error rate%	Thread throughput per second/KB	Data throudput/KB
10	0	321.1	1025
100	0	724.2	2725
1000	3.12	902.2	3542
10000	12.2	1231.1	4020

5.2 Response speed test for platform data query

Currently, the platform can provide real-time data query interfaces for single devices, real-time data query interfaces for multiple devices, and historical data query interfaces. In order to test the response speed of the data server interface, this paper uses the real-time data of one device node, the real-time data of 10 device nodes, the historical data of one device for one month, and the historical data of one device for 12 months as the test cases, and the test cases are every 100. The test yielded the average historical response time for the standard. The test results are shown in Table 2. The query speed of real-time data is slow, and the query speed of historical data increases with the increase of data volume. After optimizing the index, the query speed of historical data is obviously improved.

Table 2 Result data query response speed

Link number	cycle number	Response time /ms
Single-device-real-time	1	1321
multi-device-real-time	<100	50.6
small-scale historical	<1000	2230.6
large-scale historical	<10000	5002.2

6 Conclusions

This article mainly realizes the intelligent smart environmental protection equipment intelligently accessing the IoT cloud platform. The environmental protection equipment supports a variety of different types of equipment including PM2.5, Pm10, water and acoustic monitoring equipment etc. The transmission network uses 4G LTE and NB network technologies. The smart environmental gateway device reports various types of environmental protection data information to the equipment service management platform through the 4G LTE network or the NB-IoT network. The management platform supports intelligent access, remote management, real-time monitoring, analysis of acquired data to display and predictive alarms.

However, it is still necessary to continue research and practice in the intelligent of management platform and data transfer security in order to achieve IoT platform that is efficient, safe and convenient for marketing.

References

[1] Wang Y P E, Lin X, Adhikary A, et al. A Primer on 3GPP Narrowband Internet of Things (NB-IoT)[J]. IEEE Communications Magazine, 2016, 55(3).

[2] Mangalvedhe N, Ratasuk R, Ghosh A. NB-IoT deployment study for low power wide area cellular IoT[C]// IEEE, International Symposium on Personal, Indoor, and Mobile Radio Communications. IEEE, 2016:1-6.

[3] Ratasuk R, Vejlgaard B, Mangalvedhe N, et al. NB-IoT system for M2M communication[C]//Wireless Communications and NETWORKING Conference. IEEE, 2016:428-432.

[4] Sinha R S, Wei Y, Hwang S H. A survey on LPWA technology&58; LoRa and NB-IoT[J]. Ict Express, 2017.

[5] Hu, R., Wang, Y., and Wang, F. 2013. Environment protection based on internet of things. Applied Mechanics & Materials. 340, 993-998.

[6] Zhu, Q. 2017. On the Problems and Countermeasures in the Construction of Wisdom Environmental Protection (in Chinese). Environmental research & monitoring. 30(01), 6871.

[7] Beyene Y D, Jantti R, Ruttik K, et al. On the Performance of Narrow-Band Internet of Things (NB-IoT)[C]// Wireless Communications and NETWORKING Conference. IEEE, 2017:1-6.

[8] Guoqiang, S., Yanming, C., Chao, Z., and Yanxu, Z. 2013. Design and Implementation of a Smart IoT Gateway. Green Computing and Communications (pp.720-723). IEEE.

[9] Lea R, Blackstock M. City Hub: A Cloud-Based IoT Platform for Smart Cities[C]// IEEE, International Conference on Cloud Computing Technology and Science. IEEE, 2015:799-804.

[10] Zhao Xin. The current situation and the future development of the internet of things[J]. Computer & Network, 2012, Z1: 126-129.

[11] Jia Y J, Chen Q A, Wang S, et al. ContexIoT: Towards Providing Contextual Integrity to Appified IoT Platforms[C]// Network and Distributed System Security Symposium. 2017.