Project Report:-
Gated Recurrent Convolution Neural Network for OCR
MEASUREMENTS(NIOS 2017)

Divyansh Ahuja, 17D070038,Uzair Sayed, 17D070032

November 26, 2019

1 Introduction

OCR refers to a set of computer vision problems that require us to convert images of digital or hand-written text images to machine readable text in a form your computer can process, store and edit as a text file or as a part of a data entry and manipulation software. Unlike Recurrent Attention Models, CRNNs don't treat our OCR task as a reinforcement learning problem but as a machine learning problem with a custom loss. The loss used is called CTC loss- Connectionist Temporal Classification. The convolutional layers are used as feature extractors that pass these features to the recurrent layers - bidirectional LSTMs . These are followed by a transcription layer that uses a probabilistic approach to decode our LSTM outputs. Each frame generated by the LSTM is decoded into a character and these characters are fed into a final decoder/transcription layer which will output the final predicted sequence.

2 Architecture

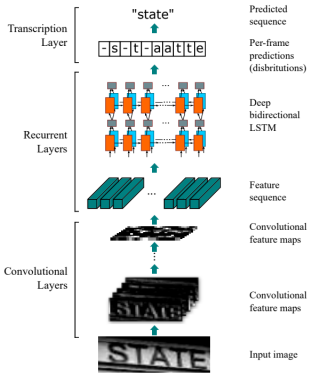

Figure 1. The network architecture. The architecture consists of three parts: 1) convolutional layers, which extract a feature sequence from the input image; 2) recurrent layers, which predict a label distribution for each frame; 3) transcription layer, which translates the per-frame predictions into the final label sequence.

Source: Jianfeng Wang GRCNN paper

2.1 GRCL

This module is equipped with a gate to control the context modulation in RCL and it can weaken or even cut off some irrelevant context information. This is because, with increasing iterations, the size of the effective receptive field will increase unboundedly, which contradicts the biological fact. One needs a mechanism to constrain the growth of the effective receptive field. In addition, from the viewpoint of performance enhancing, one also needs to control the context modulation of neurons in RCNN.

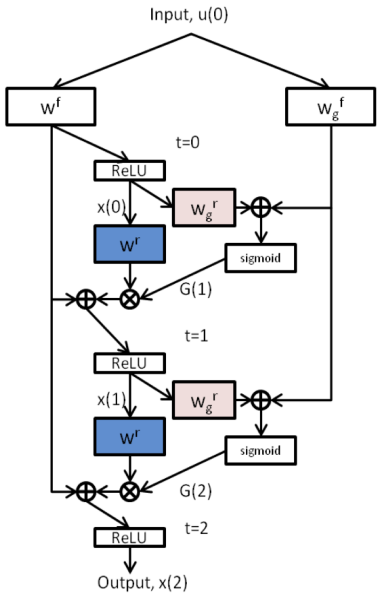

Figure 2: Illustration of GRCL with $T = 2$. The convolutional kernels in the same color use the same weights.

3 Feature Sequence Extraction

The input to the network is a whole image and the image is resized to fixed length and height. The feature map in the last layer is sliced from left to right by column to form a feature sequence. Therefore, the i-th feature vector is formed by concatenating the i-th columns of all of the maps. Each feature vector generated by the GRCL represents a larger region. This feature vector contains more information than the feature vector generated by the basic convolutional layer, and it is beneficial for the recognition of text.

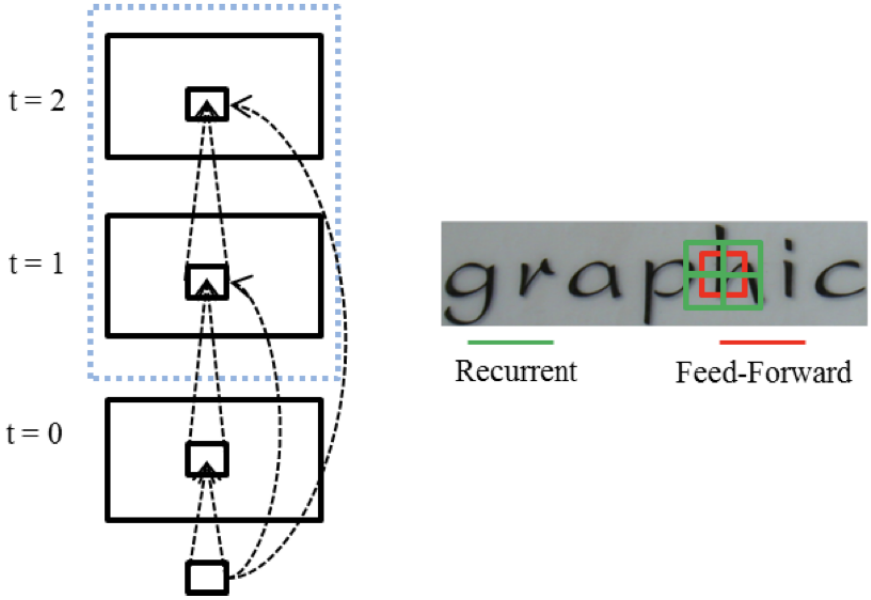

Figure 1: Illustration of using RCL with $T = 2$ for OCR.

4 Sequence Modeling

An LSTM is used on the top of feature extraction module for sequence modeling.
γ_i is defined as an indication factor and its value is 0 or 1. When γ_i is equal to 1, the gate receives the modulation of the cell's state.
However, LSTM only considers past events. In fact, the context information from both directions are often complementary. Therefore, we use stacked bidirectional LSTM in our architecture.

5 Transcription

We use a lexicon-free transcription, which converts per-frame predictions to real labels without the use of a dictionary Given an input image I, the prediction of RNN at each time step is denoted by πt.The Connectionist Temporal Classification (CTC) method is used The sequence π may contain blanks and repeated labels (e.g. (a b b)=(ab bb)) and we need a concise representation l (the two examples are both reduced to abb).

6 Implementation Details

The following is the GRCNN configuration used.

Table 1: The GRCNN configuration

Conv	MaxPool	GRCL	MaxPool	GRCL	MaxPool	GRCL	MaxPool	Conv
3×3	2×2	3×3	2×2	3×3	2×2	3×3	2×2	2×2
num: 64		num: 64		num: 128		num: 256		num: 512
sh:1 sw:1	sh:2 sw:2	sh:1 sw:1	sh:2 sw:2	sh:1 sw:1	sh:2 sw:1	sh:1 sw:1	sh:2 sw:1	sh:1 sw:1
ph:1 pw:1	ph:0 pw:0	ph:1 pw:1	ph:0 pw:0	ph:1 pw:1	ph:0 pw:1	ph:1 pw:1	ph:0 pw:1	ph:0 pw:0

Before input to the network, the pixel values are rescaled to the range (-1, 1). The final output of the feature extractor is a feature sequence of 26 frames. The recurrent layer is a bidirectional LSTM with 512 units without dropout. The Adam's optimizer is used for training. We trained the proposed model on IC03 dataset, which contains 860 character sequence images, divided it into 760 training images and 100 testing images, the obtained accuracy on test set and comparision with other methods is as follows:

Method	Dataset IC03
Jaderberg et al.	89.6%
Baoguang et al.	89.4%
Chen-Yu et al.	88.7%
ResNet-BLSTM	89.9%
Jianfeng Wang	91.2%
Our Implementation	Train(95.7%) Test(88%)

7 Conclusion

The paper proposed a new architecture named GRCNN which uses a gate to modulate recurrent connections in a previous model RCNN. GRCNN is able to choose the context information dynamically and combine the feed-forward part with recurrent part flexibly. The unrelated context information coming from the recurrent part is inhibited by the gate.

8 Refrences

This work was done as a course project of GNR638 at IIT Bombay.
1. Jianfeng Wang and Xiaolin Hu. Gated Recurrent Convolution Neural Network for OCR
2. A. Graves and F. Gomez. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks.