Training an End-to-end Model for Offline Handwritten Japanese Text Recognition by Generated Synthetic Patterns

Nam-Tuan Ly, Cuong-Tuan Nguyen, Masaki Nakagawa

Tokyo University of Agriculture and Technology

Outline

1. Introduction
2. Proposed method
3. Experiments
4. Conclusion & Future Work

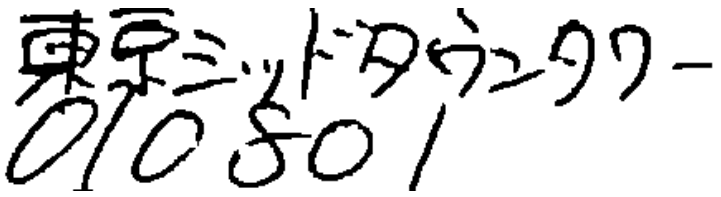
Outline

1. Introduction
2. Proposed method
3. Experiments
4. Conclusion & Future Work

Background

- Offline handwritten Japanese text recognition:
  - Big challenging problem.
  - Receiving much attention from numerous business sectors.
- The existing systems are still far from perfection:
  - Thousands of classes (4,438 classes) and various characters: Kana, Kanji, numerals and alphabet characters.
  - Diversity of writing styles.
  - Multiple-touches between characters.
  - Noises...
- Handwritten Japanese text database, TUAT Kondate:
  - 13,685 text line images.
  - Covers ~1200 categories characrers
  - → Data is not enough.

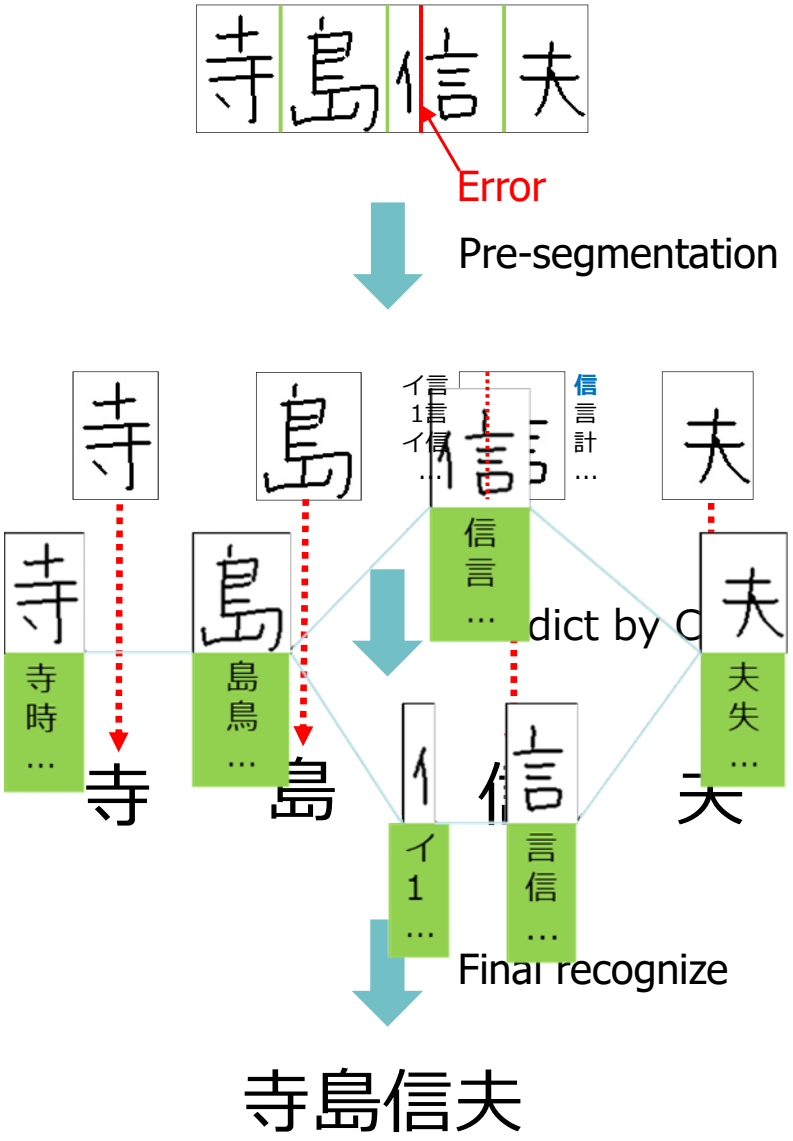
Samples of Japanese characters



Handwritten Japanese text

Related Work(1/3)

- Segmentation based methods (*).
  - Pre-segment text lines into characters.
  - Individually recognize each character by MQDF or CNN.
  - Finally recognize text lines while integrating linguistic and geometric contexts.
  - They were dominant for Japanese.
  - → Problems:
    - ✓ Pre-segmentation of text lines is quite costly.
    - ✓ Early errors have domino-effect on the performance.

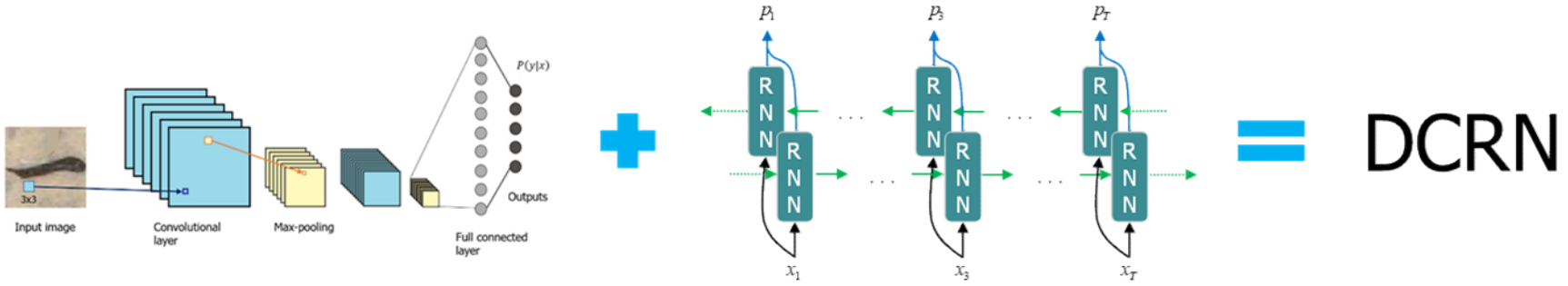


(*) *K. C. Nguyen and M. Nakagawa 2016, Q.-F. Wang et al 2012, S. N. Srihari et al 2007.*

Related Work(2/3)

- Segmentation-free methods: avoiding segmentation errors.
  - Traditional segmentation-free methods are HMM-based (*).
    - → Deep Neural Nets have proven superior to HMM.
  - Based on Deep Neural Nets and CTC, many segmentation-free methods have been proposed and proven to be very powerful.
    - Graves et al. (2009) combined BLSTM and CTC to build a Connectionist System.
    - R. Messina et al. (2015) combined MDLSTM-RNN and CTC to build an end-to-end trainable model.

→We propose an end-to-end model of Deep Convolutional Recurrent Network (DCRN) for offline handwritten Japanese text recognition.



CNN RNN

(*) *Su et al., 2009, Suryani et al 2016.*

Related Work(3/3)

- Deep Neural Networks typically require a large set of data for training.
  - ◆ Handwritten Japanese Text dataset, TUAT Kondate: data is not enough.
  - → apply data argumentation.
- Many data argumentation methods have been proposed by modifying the original patterns:
  - ◆ Affine transformations, nonlinear combinations...
  - → However, such methods just modify the original patterns, can't gain the real text line images.
- We propose a synthetic pattern generation method.

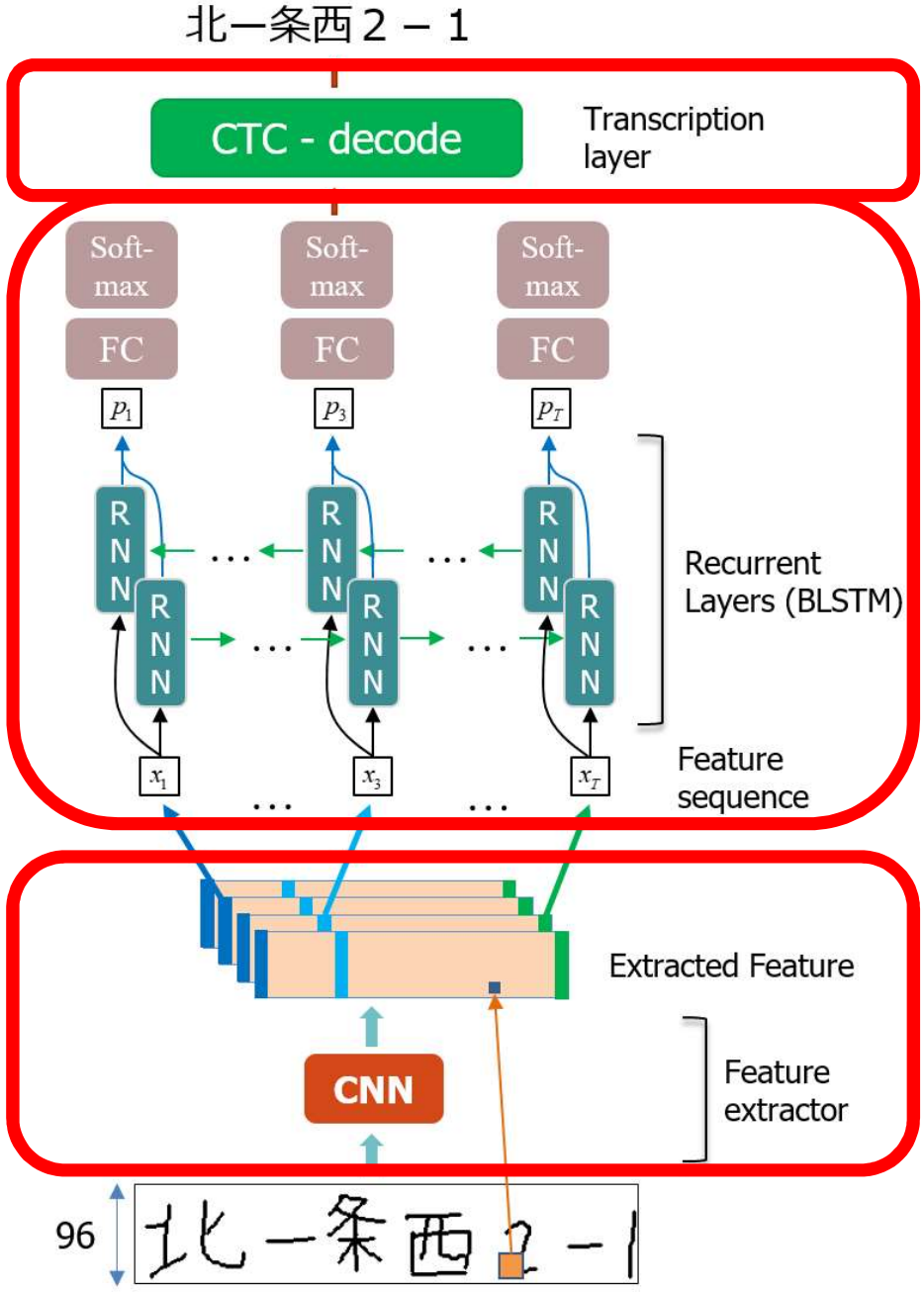
Outline

1. Introduction
2. Proposed method
3. Experiments
4. Conclusion & Future Work

Deep Convolutional Recurrent Network(1/3)

Deep Convolutional Recurrent Network (DCRN) consists of three components.

- Convolutional Feature Extractor.
  - Using a standard CNN network (FC and Softmax layers are removed).
  - Extract a feature sequence from a text line image.
- Recurrent layers.
  - Employing a Bidirectional LSTM.
  - Predict pre-frames from a feature sequence.
- Transcription layer.
  - Using CTC – decoder.
  - Convert the pre-frame predictions into a label sequence.

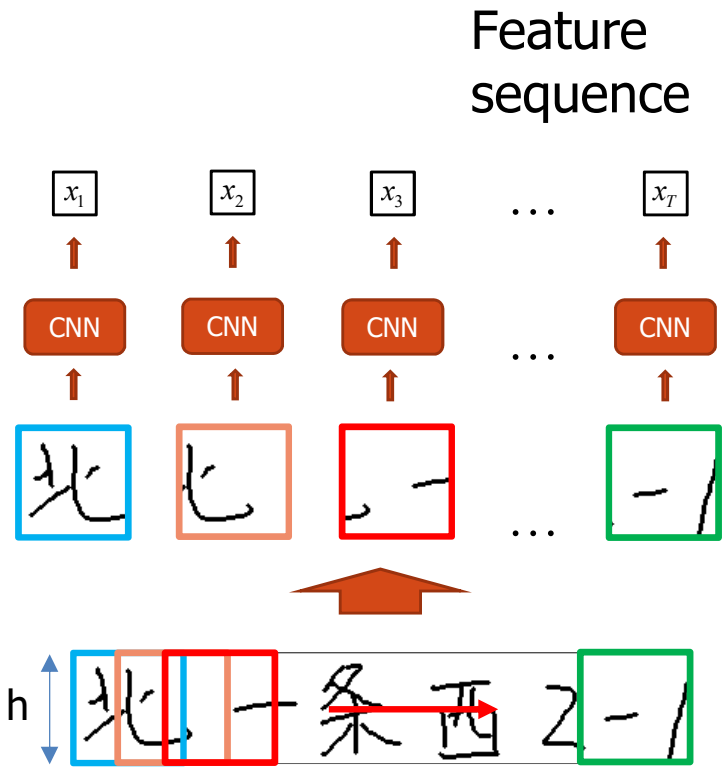


Deep Convolutional Recurrent Network(2/3)

Previous works(*): overlapped sliding windows DCRN model

Convolutional Feature Extractor.

- Pretrain CNN by isolated character patterns.
- Using the pretrained CNN and overlapped sliding windows to extract a feature sequence.



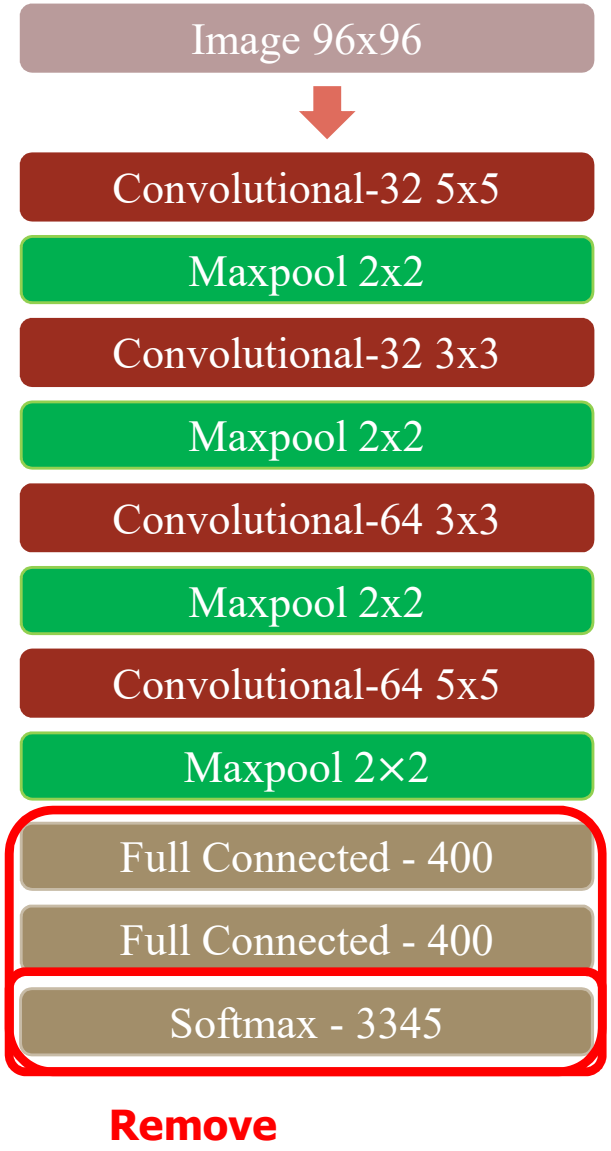
(*) Nam-Tuan Ly et al. 2017

Deep Convolutional Recurrent Network(2/3)

Previous works(*): overlapped sliding windows DCRN model

Convolutional Feature Extractor.

- Pretrain CNN by isolated character patterns.
- Using the pretrained CNN and overlapped sliding windows to extract a feature sequence.
- Two configurations:
  - Remove just Softmax layer from CNN.
    →**DCRN_o-s**
  - Remove both FC and Softmax layers from CNN.
    →**DCRN_o-f&s**



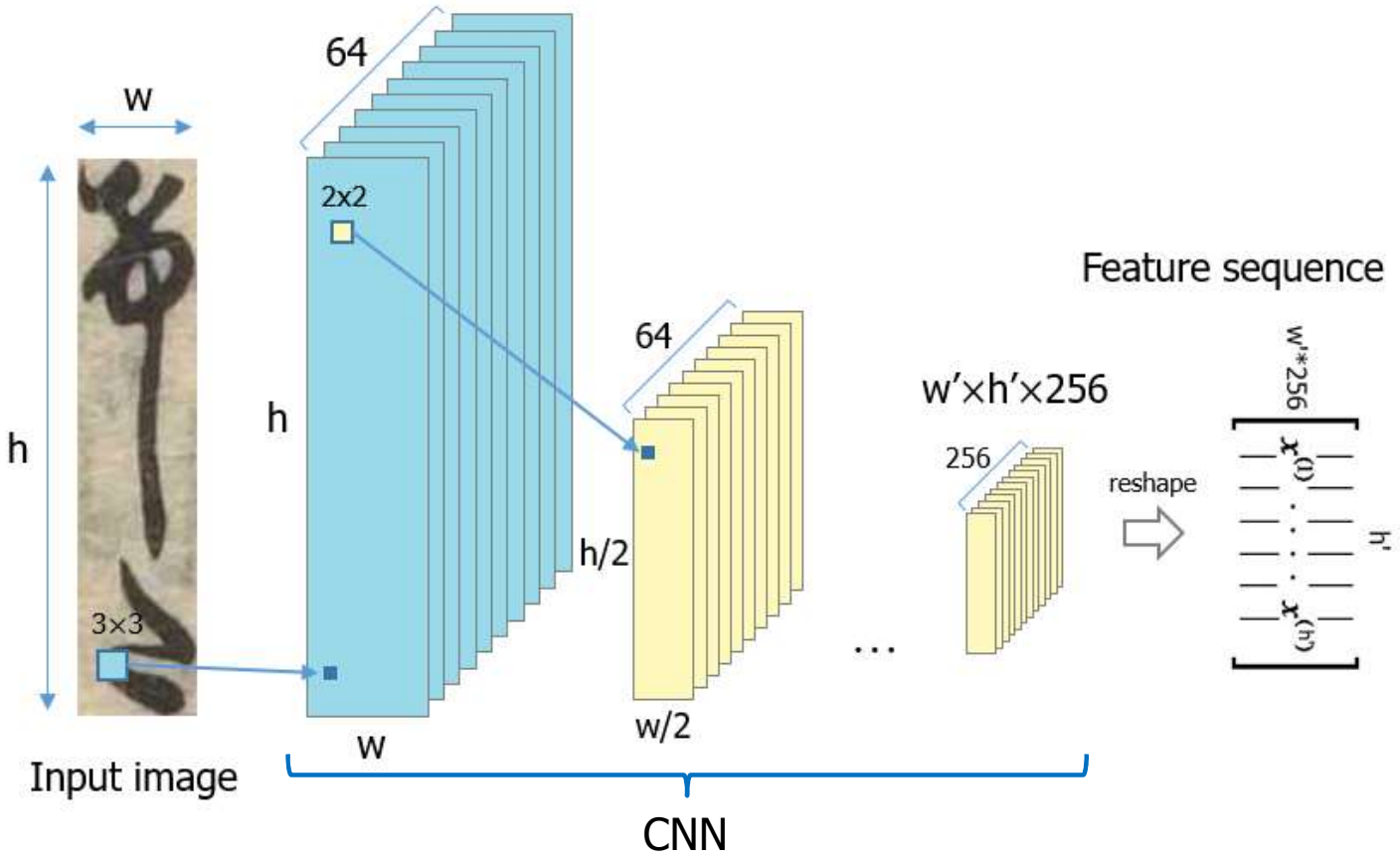
(*) Nam-Tuan Ly et al. 2017

Deep Convolutional Recurrent Network(3/3)

This works: End-to-end Model

- Remove softmax and FC layers from CNN.
- Do not use sliding windows.
- Do not pretrain CNN.
- End-to-end training System.

→ **End-to-End**



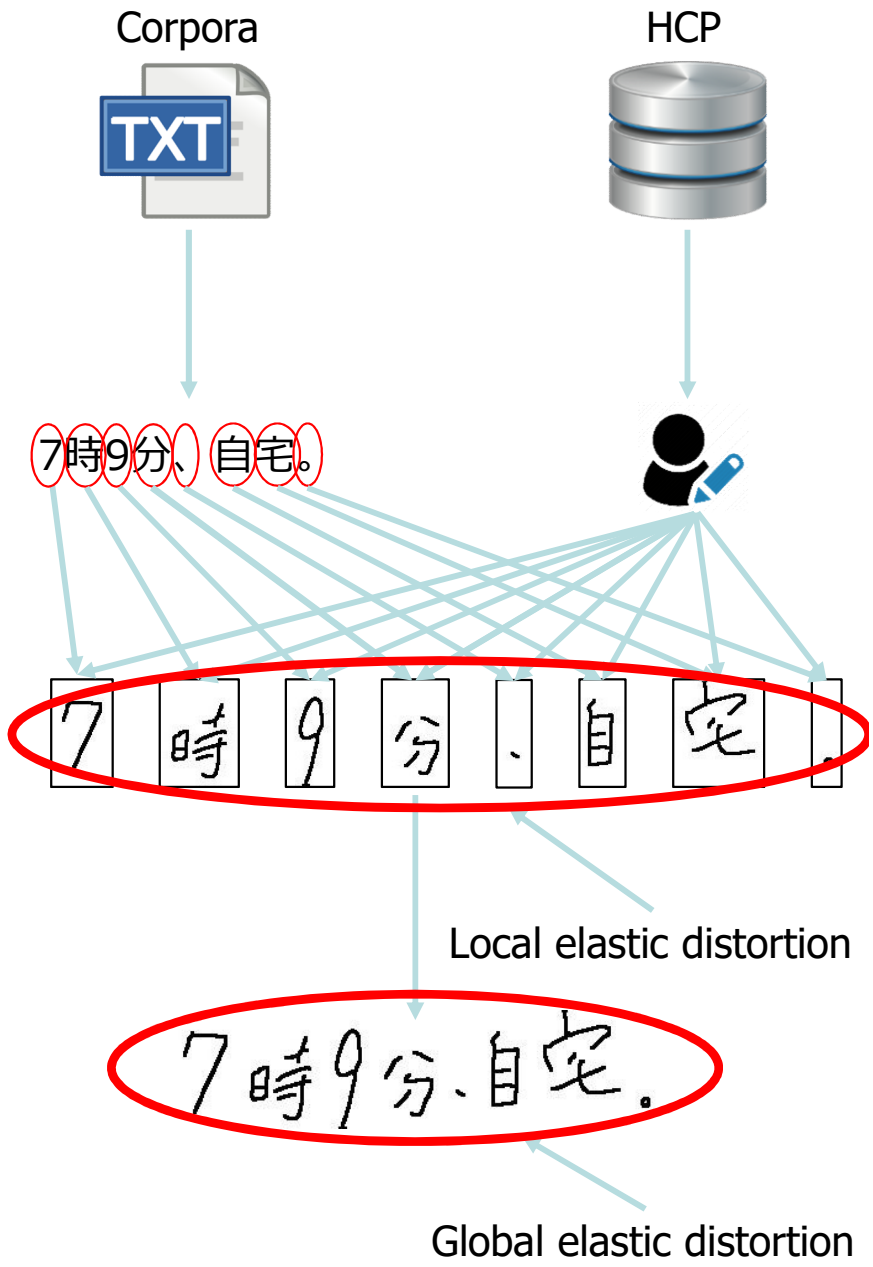
Synthetic Data Generation(1/3)

Synthetic pattern generation method.

- ◆ Sentences in corpora and handwritten character pattern database (HCP).
- ◆ Local and global elastic distortion model.

□ Following 6 steps:

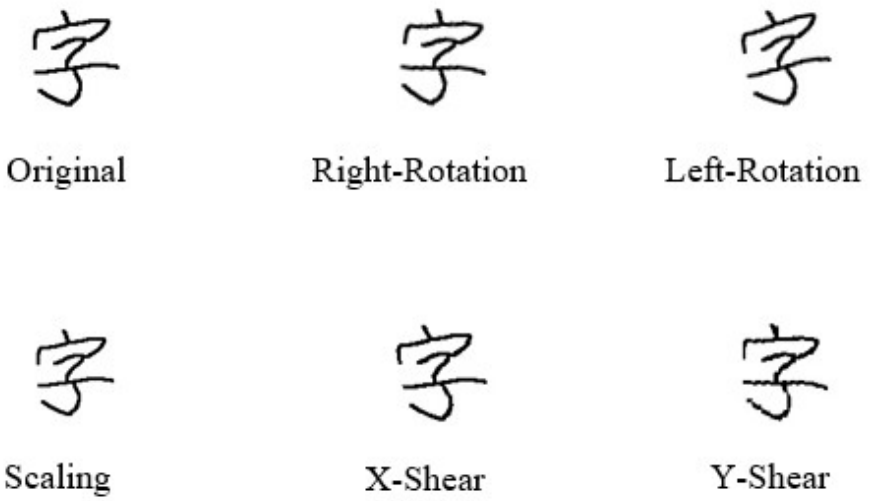
1. Get a sentence (S) from a corpus.
2. Randomly choose a writer (A) from the HCP.
3. For each character of the sentence (S), a handwritten image of this character is randomly chosen from the writer (A).
4. Apply a local elastic distortion to each handwritten character pattern in the step 3.
5. Synthesize a handwritten text line image from the sentence (S) and handwritten character images in the step 4 with random spacing.
6. Apply a global elastic distortion to the generated synthetic text line image.



Synthetic Data Generation(2/3)

Local Elastic Distortion

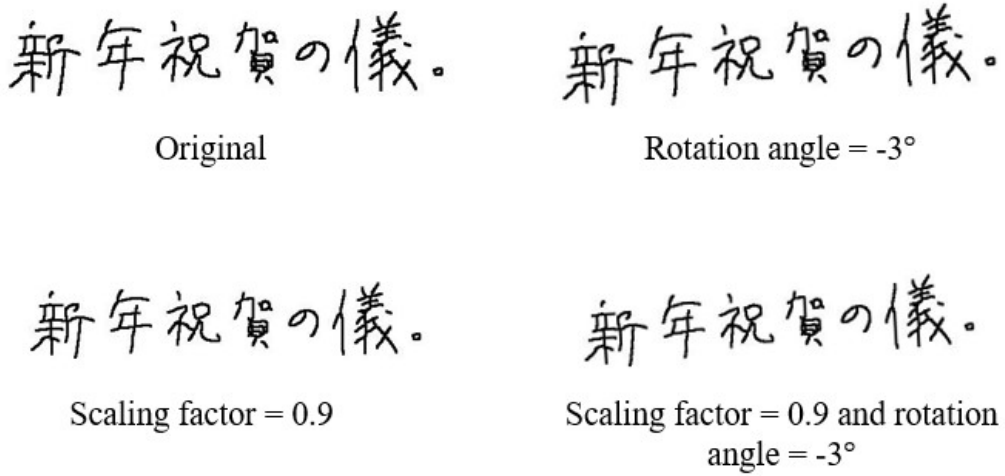
- Performs affine transformations on each handwritten character image.
- Employs the shearing, rotation, scaling, translation transformations.



Local elastic distortion

Global Elastic Distortion

- Performs affine transformations on a whole text line image.
- Employs the rotation and scaling transformations.



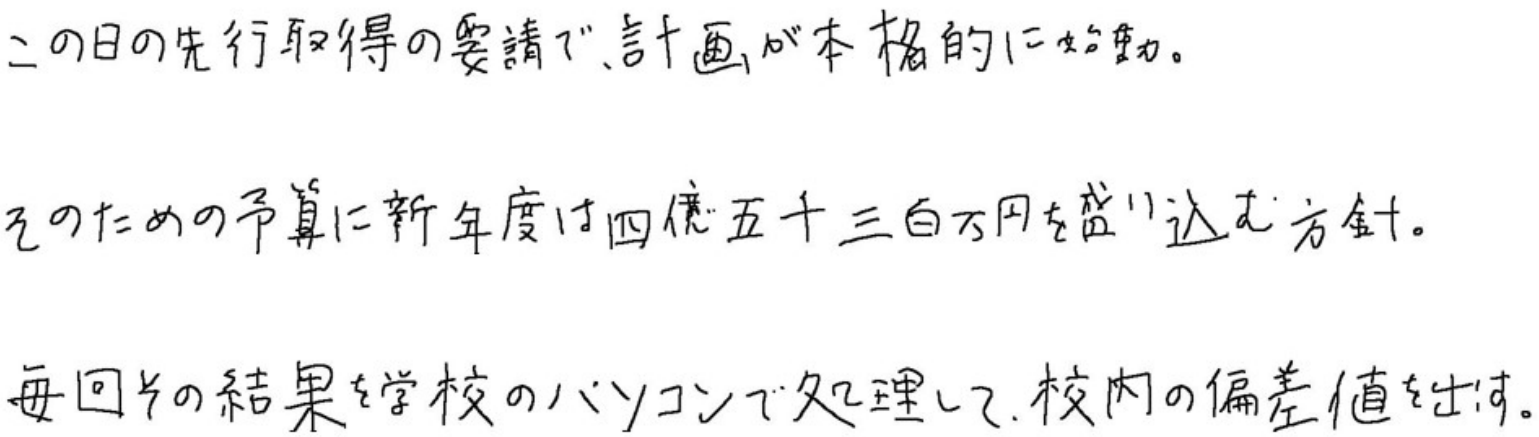
Global elastic distortion

Synthetic Data Generation(3/3)

Synthetic Handwritten Text Line Dataset (SHTL)

- Handwritten Japanese character pattern DBs, Nakayosi and Kuchibue.
- Nikkei newspaper corpus (1.1 million sentences) and Asahi newspaper corpus (1.14 million sentences).
  - ◆ Randomly choose 30,000 sentences which contain less than 30 characters from each corpus.

→ make sure that the end-to-end model can be trainable by SHTL.



Samples of generated synthetic data.

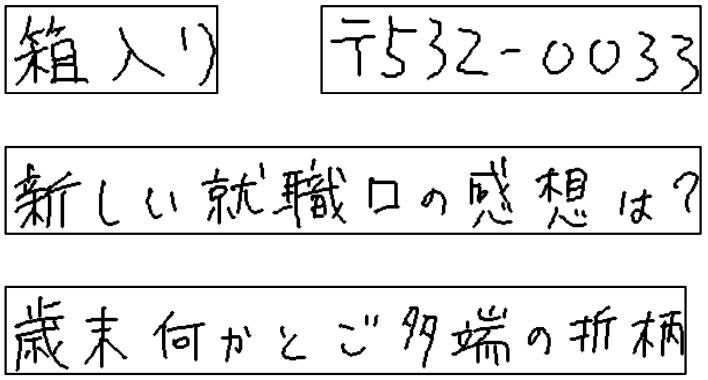
Outline

1. Introduction
2. Proposed method
3. Experiments
4. Conclusion & Future Work

Datasets(1/2)

TUAT Kondate database

- A database of handwritten text patterns mixed with figures, tables, maps, diagrams and so on (originally online but converted to offline).
  - About 13,685 of text line patterns (from 100 Japanese writers).



Kondate sample patterns.

Information on Kondate database

	Kondate		
	Train set	Valid set	Test set
Number of writers	84	6	10
Number of samples	11,487	800	1,398

Datasets(2/2)

Handwritten Japanese character pattern database.

- Nakayosi & Kuchibue (originally online but converted to offline)
  - Used for generating SHTL.

	Nakayosi	Kuchibue
Writers	163	120
Classes	4438	3345
Samples	1,695,689	1,435,440

Synthetic Handwritten Text Line Dataset (SHTL)

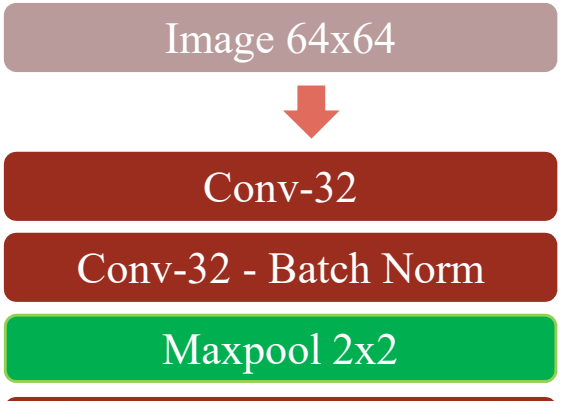
- 60,000 text line images.

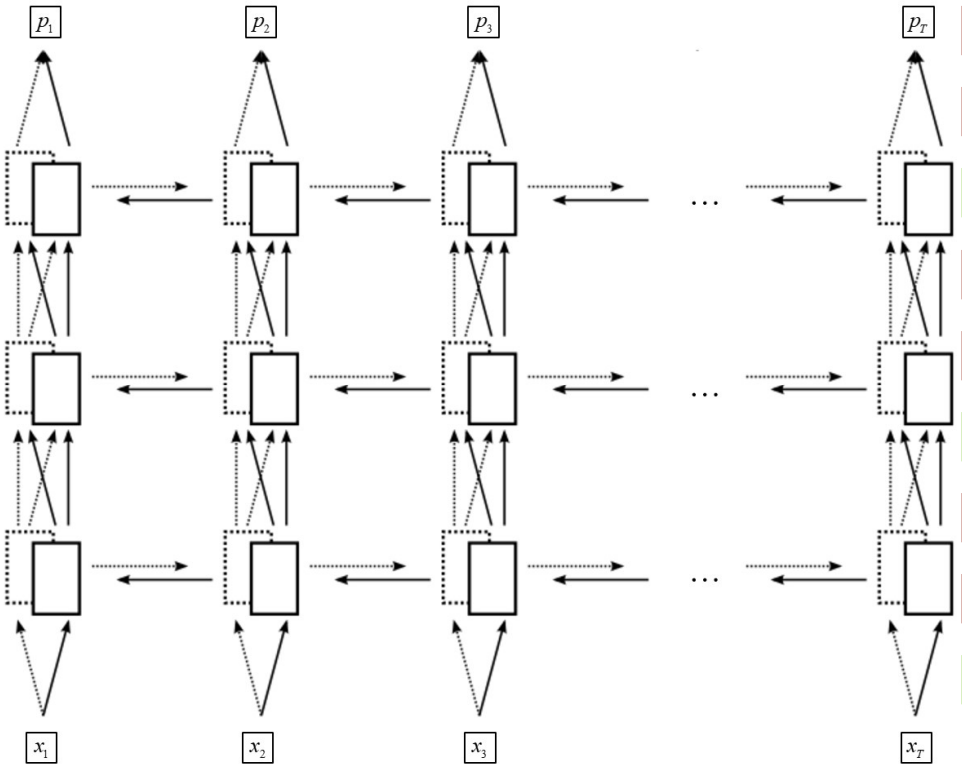
→ used for training the end-to-end model.

Implementation Details

End-to-end DCRN

- Convolutional Feature Extractor: CNN network.
  - 4 cascades of 2 convolutional and pooling layers.
  - Batch normalization, Leaky ReLu.
- Recurrent layers: Deep BLSTM.
  - Three layers of 128 nodes each.
  - Dropout (dropout rate = 0.8).
  - FC and Softmax layers.
- Training by 2 datasets:
  - TUAT Kondate
    → **End-to-End**
  - TUAT Kondate + SHTL
    → **End-to-End_SHTL**





Evaluation Results

- Label Error Rate (LER)

$$LER(h, S') = \frac{1}{Z} \sum_{(x,z) \in S'} ED(h(x), z)$$

- Sequence Error Rate (SER):

$$SER(h, S') = \frac{100}{|S'|} \sum_{(x,z) \in S'} \begin{cases} 0 & if \quad h(x) = z \\ 1 & otherwise \end{cases}$$

- Where Z is the total number of target labels in S'
- ED(p, q) is the edit distance between two sequences p and q.

Experiment Results

Label Error Rate (LER) and Sequence Error Rate (SER) on TUAT Kondate dataset.

Model	Label Error Rate(%)		Sequence Error Rate(%)	
	Valid set	Test set	Valid set	Test set
DCRN_o-f&s	11.74	6.95	39.33	28.04
DCRN_o-s	11.01	6.44	37.38	25.89
End-to-End	5.22	**3.65**	24.47	**17.24**
End-to-End_SHTL	3.62	**1.95**	21.87	**14.02**

- The end-to-end DCRN models substantially work better than the overlapped sliding windows DCRN model.
- Recognition accuracy is improved by using the SHTL dataset for training the end-to-end model.

Experiment Results

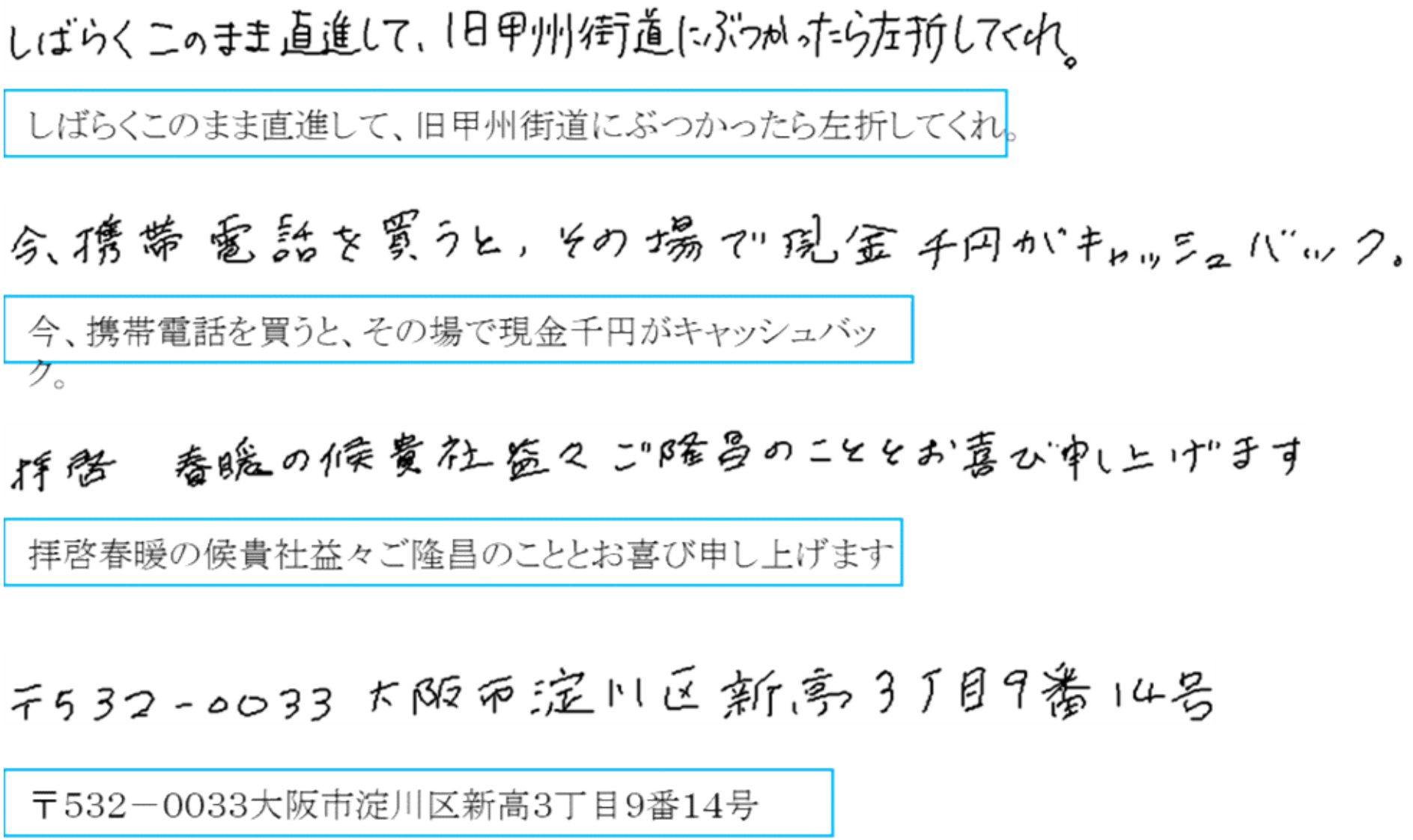
Label Error Rate and Sequence Error Rate when combined with the language model.

Model	Test set	
	LER(%)	SER(%)
Segmensation based [1]	11.2	48.53
DCRN_o-f&s	6.68	26.97
DCRN_o-s	6.10	24.39
End-to-End	**3.52**	**16.67**
End-to-End_SHTL	**1.87**	**13.81**

[1] K. C. Nguyen et al.

- The DCRN models are superior to the segmentation based method.
- Recognition accuracy is further improved when the linguistic context is integrated.

Correctly recognized samples



Correctly recognized samples by End-to-End_SHTL.

Misrecognized samples

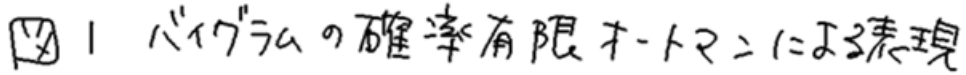
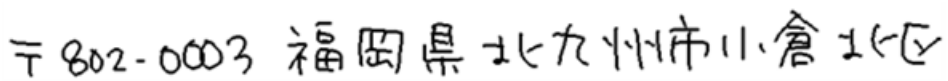
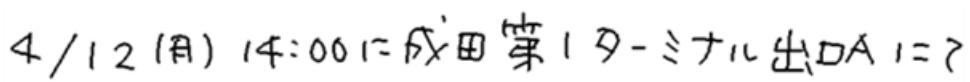


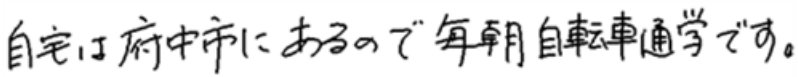
図1バイグラムの確率有限オートマンによる表現 -> 図1バイグラムの確率有限オートマンによる現



〒802−0003福岡県北九州市小倉北区 -> 〒002−0003福岡県北九州市小倉北区



4／12(月)14:00に成田第1ターミナル出口Aにて -> 4／12(月)14:00に成田第1ターミナル出口Aに？



自宅は府中市にあるので毎朝自転車通学です。 -> 自宅は府中市にあるので毎朝自東車画学です。

Some mispredicted samples by End-to-End_SHTL.

Outline

1. Introduction
2. Proposed method
3. Experiments
4. Conclusion & Future Work

Conclusion

- The end-to-end DCRN models substantially outperform the overlapped sliding windows model and the segmentation-based method.
- The synthetic pattern generation method improves the accuracy of the end-to-end DCRN models.
- Recognition rate is further improved when combined with the language model.

Future Work

- Apply the DCRN model for offline handwritten multi-lines data.
- Apply the RNN language model and compare it with the tri-gram language model.
- Apply for the JIS level 2 characters (~7,000 categories).

Thank you for your attention.