



- (51) **International Patent Classification:**
G06K 9/18 (2006.01)
- (21) **International Application Number:**
PCT/CN20 15/08 1308
- (22) **International Filing Date:**
12 June 2015 (12.06.2015)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (71) **Applicant: SENSETIME GROUP LIMITED** [CN/CN];
Unit 319, 3/F, Building 16W, No. 16 Science Park West
Avenue, HKSTP, Shatin, Hong Kong (CN).
- (72) **Inventors: TANG, Xiaoou;** Department of Information
Engineering, The Chinese University of Hong Kong, Shat-
in, Hong Kong (CN). **HUANG, Weilin;** Shenzhen Insti-
tutes of Advanced Technology, Chinese Academy of Sci-
ences, Shenzhen, Guangdong (CN). **QIAO, Yu;** Shenzhen
Institutes of Advanced Technology, Chinese Academy of
Sciences, Guangdong (CN). **LOY, Chen Change;** Depart-
ment of Information Engineering, The Chinese University
of Hong Kong, Shatin, Hong Kong (CN). **HE, Pan;** Shen-
zhen Institutes of Advanced Technology, Chinese
Academy of Sciences, Shenzhen, Guangdong (CN).

- (74) **Agent: INSIGHT INTELLECTUAL PROPERTY LIM-
ITED;** 19 A, Tower A, InDo Building, No. 48A Zhichun
Road, Haidian District, Beijing 100098 (CN).
- (81) **Designated States** (*unless otherwise indicated, for every
kind of national protection available*): AE, AG, AL, AM,
AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY,
BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM,
DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT,
HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR,
KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG,
MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM,
PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC,
SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN,
TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) **Designated States** (*unless otherwise indicated, for every
kind of regional protection available*): ARIPO (BW, GH,
GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ,
TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU,
TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE,
DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU,
LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK,
SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ,
GW, KM, ML, MR, NE, SN, TD, TG).

Published:
— with international search report (Art. 21(3))

(54) **Title:** METHODS AND APPARATUS FOR RECOGNIZING TEXT IN AN IMAGE

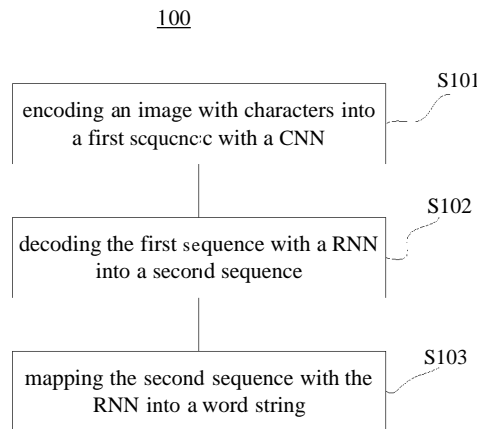
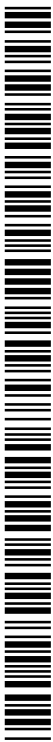


Figure 1

(57) **Abstract:** Methods and apparatus for recognizing text in an image are disclosed. According to an embodiment, the method comprises encoding the image into a first sequence with a convolutional neural network (CNN), wherein the first sequence is an output from a last second convolutional layer of the CNN; decoding the first sequence with a recurrent neural network (RNN) into a second sequence, which has a same length as the first sequence and indicates estimated probabilities over all possible characters corresponding to each component in the first sequence; and mapping the second sequence into a word string in which repeated labels and non-character labels are removed.



Methods and Apparatus for Recognizing Text in an Image

Technical Field

[0001] This application relates to text recognizing; in particular, to methods and apparatus for recognizing text in an image.

Background

[0002] Text recognition in natural image has received increasing attention in computer vision, due to its numerous practical applications. This problem includes two sub tasks, namely text detection and text-line/word recognition. The main difficulty arises from the large diversity of text patterns (e.g. low resolution, low contrast, and blurring), and highly complicated background clutters. Consequently, individual character segmentation or separation is extremely challenging.

[0003] Most previous studies focus on developing powerful character classifiers, some of which are incorporated with an additional language model, leading to the state-of-the-art performance. These approaches mainly follow the basic pipeline of conventional OCR techniques by first involving a character-level segmentation, then followed by an isolated character classifier and post-processing for recognition. Several approaches adopt deep neural networks for representation learning, but the recognition is still confined to character-level classification. All the current successful scene text recognition systems are mostly built on isolated character classifier. Their performance are thus severely harmed by the difficulty of character-level segmentation or separation. Importantly, recognizing each character independently discards meaningful context information of the words, significantly reducing its reliability and robustness.

Summary

[0004] According to an aspect of the present application, a method for recognizing text in an image comprises: encoding the image into a first sequence with a convolutional neural network (CNN), wherein the first sequence is an output from a last second convolutional layer of the CNN; decoding the first sequence with a

recurrent neural network (RNN) into a second sequence, which has a same length as the first sequence and indicates estimated probabilities over all possible characters corresponding to each component in the first sequence; and mapping the second sequence into a word string in which repeated labels and non-character labels are removed.

[0005] According to another aspect of the present application, an apparatus for recognizing text in an image comprises: a convolutional neural network (CNN) encoding the image into a first sequence, wherein the first sequence is an output from a last second convolutional layer of the CNN; and a recurrent neural network (RNN) decoding the first sequence into a second sequence, which has a same length as the first sequence and indicates estimated probabilities over all possible characters corresponding to each component in the first sequence; wherein the RNN further maps the second sequence into a word string in which repeated labels and non-character labels are removed.

Drawings

[0006] Fig. 1 is flowchart of a method for recognizing text in an image according to an embodiment of the present application.

[0007] Fig. 2 illustrates an overall pipeline of an apparatus or a system for recognizing text in an image according to an embodiment of the present application.

[0008] Fig. 3 illustrates a structure of a five-layer Maxout CNN mode used in an embodiment of the present application.

[0009] Fig. 4 illustrates a structure of a RNN mode used in an embodiment of the present application.

[0010] Fig. 5 illustrates a structure of the memory cells in a RNN mode used in an embodiment of the present application.

Detailed Description

[0011] Embodiments of the present application are described below with reference to the accompanying drawings in detail.

[0012] Fig. 1 illustrates a flowchart of a method 100 for recognizing text in an image according to an embodiment of the present application. As shown in Fig. 1, at step S101, an image with characters is encoded into a first sequence with a convolutional neural network (CNN), wherein the first sequence is an output from a last second convolutional layer of the CNN. At step S102, the first sequence is decoded with a recurrent neural network (RNN) into a second sequence, which has a same length as the first sequence and indicates estimated probabilities over all possible characters corresponding to each component in the first sequence. At step S103, the second sequence is mapped by into a word string in which repeated labels and non-character labels are removed.

[0013] According to the embodiment, no character segmentation is needed. Instead, the output from the last second convolutional layer of the CNN is obtained and directly used as an input of the RNN for text recognizing, so that an advantage of the RNN in retaining meaningful interdependencies of continuous text is considered during the process.

[0014] In an embodiment, the CNN may perform convolution to the image as a whole. In this case, the result of the convolution obtained from the last second convolutional layer of the CNN is the first sequence that is to be used as the input of the RNN.

[0015] Alternatively, the CNN may use a slide window to scan the word image from left to right densely and divide the image into continuous segments. Note that, such segments are not equal to those obtained by character segmentation, since it is just simply scanning and dividing, without needing to identify individual characters. The segments are convoluted individually by the CNN. The results of the convolution obtained from the last second convolutional layer of the CNN are components which collectively form the first sequence that is to be used as the input of the RNN.

[0016] Fig. 2 illustrates an overall pipeline of an apparatus or a system for recognizing text in an image according to an embodiment of the present application. As shown in the top box, an image with a character string "apartment" is divided into

segments of the same size, e.g., 32*32. Such division is implemented with a sliding window by scanning. The division involves no character identifying work. Each segment may comprise one or more complete or incomplete characters, and may comprise no character at all, as can be seen from the second row in the top box.

[0017] The middle box in Fig. 2 shows a CNN network, which performs convolution to each of the segment sequentially. As is known, a CNN may comprise several layers. In prior art, some text recognizing technologies have used the CNN network for an isolated character recognition , in which a character label is output at the last layer of the CNN. According to the present application, the output from the last second layer of the CNN is used. The output has 128 feature maps, each of which includes a single neuron. For example, a segment with a size of 32*32 may generate a 128D output from the last second layer of the CNN. For an image which may be divided by a sliding window into T segments, T number of 128D outputs may be generated, wherein T is a positive integer varied by the aspect ratio of the image.. The output sequence represents high-level deep features of the input image.

[0018] The bottom box in Fig. 2 shows an RNN network, which decodes the output sequence from the CNN. As can be seen, the RNN has the same number of channels as the CNN. However, different from the CNN in which each channel works individually, the sequential channels in the RNN are connected and interacted by internal states of the RNN in the hidden layers. Based on such configuration, for each component in the sequence output from the CNN, estimated probabilities over all possible characters are output, with consideration of the relationship with both the previous component (if any) and the following component (if any) in the sequence. The estimated probabilities for each component in the sequence output from the CNN (thus for each segment of the image) are then considered together and mapped into a word string "apartment" , in which repeated labels and non-character labels are removed.

[0019] According to an embodiment, the image may be resized to be adaptive to the CNN or the sliding window so that it can be properly processed and recognized.

For example, for a sliding window having a size of 32*32, the image may be resized to have a height of 32, while keeping its original aspect ratio unchanged.

[0020] Although Fig. 2 shows a situation when a sliding window is used. In another embodiment, the sliding window is not necessary. Under such circumstances, the image is convoluted as a whole, and the output from the last second layer of the CNN has a matrix of 128*T, which is equivalent to the sequence obtained by connecting T 128D outputs together.

[0021] Before describing the CNN and RNN in detail, the process of word image recognition is formulated as a sequence labeling problem as below. The probability of the correct word strings (S_w) is maximized given an input image (I) as follows,

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} \sum_{(I, S_w)} \log P(S_w | I; \theta) \tag{1}$$

where θ are the parameters of the recurrent system. $(I, S_w) \in \Omega$ is a sample pair from a training set, Ω , where $S_w = \{S_w^1, S_w^2, \dots, S_w^K\}$ is the ground truth word string (containing K characters) of the image I. Commonly, the chain rule is applied to model the joint probability over S_w ,

$$\log P(S_w | I; \theta) = \sum_{i=1}^K \log P(S_w^i | I, S_w^0, \dots, S_w^{i-1}; \theta) \tag{2}$$

[0022] Thus the sum of the log probabilities over all sample pairs in the training set (Ω) is optimized to learn the model parameters. A RNN is developed to model the sequential probabilities $P(S_w^i | I, S_w^0, \dots, S_w^{i-1})$, where the variable number of the sequentially conditioned characters can be expressed by an internal state of the RNN in hidden layer, h_t . This internal state is updated when the next sequential input x_t is presented by computing a non-linear function H ,

$$h_{t+i} = H(h_t, x_t) \tag{3}$$

where the non-linear function H defines exact form of the proposed recurrent

system. $X = \{x_1, x_2, \dots, x_T\}$ is the sequential CNN features computed from the word image.

$$\{x_1, x_2, \dots, x_T\} = \phi(I) \quad (4)$$

[0023] Designs of the ϕ and H play crucial roles in the proposed system. A CNN model is developed to generate the sequential x_i , and define H with a long short-term memory (LSTM) architecture.

[0024] Both CNN and RNN are previously trained, which will be described in detail as below.

[0025] To better understand and implement embodiments of the application, a five-layer maxout CNN and a bidirectional long short-term memory (LSTM) based on RNN are used in the following illustrative example. The LSTM-based RNN may further comprise a connectionist temporal classification (CTC) layer. It is noted that other kinds of CNN and/or RNN are also possible to be used to implement the application.

[0026] Fig. 3 illustrates a structure of a five-layer Maxout CNN mode used in an embodiment of the present application. As shown, the basic pipeline is to compute point-wise maximum through a number of grouped feature maps or channels. For example, an input image/segment is of a size 32*32 corresponding the size of sliding window. The Maxout CNN network has five convolutional layers, each of which is followed by a two-group or four-group Maxout operation, with different numbers of feature maps, i.e. 48, 64, 128, 128 and 36 respectively. During the convolution, no pooling operation is involved, and the output map of last two convolutional layers are just one pixel. This allows the CNN to convolute the whole word images at once, leading to a significant computational efficiency. For each word image, it may be resized into the same height of 32, and keep its original aspect ratio unchanged. By applying the learned CNN to the resized image, a 128D CNN sequence may be directly from the output of last second convolutional layer. This operation is similar to computing deep feature independently from the sliding window by moving it densely

through the image, but with much computational efficiency. The used Maxout CNN may be trained on 36 classes comprised of case insensitive character sample images, including 26 characters and 10 numbers.

[0027] As mentioned above, for an image divided by a sliding window into T segments, the output from the CNN to the RNN is a matrix $\hat{X} = \{x_1, x_2, \dots, x_T\}$, which each of x_1, x_2, \dots, x_T is a 128D vector.

[0028] Fig. 4 illustrates a structure of a RNN mode used in an embodiment of the present application. The RNN comprises bidirectional long short-term memory (LSTM) layer and a connectionist temporal classification (CTC) layer, wherein the LSTM layer generates the second sequence from the input CNN sequence, and the CTC layer generates the word string from the second sequence.

[0029] As shown, the bidirectional LSTM has two separate LSTM hidden layers that process the input sequence forward and backward, respectively. Both hidden layers are connected to the same input and output layers.

[0030] The main shortcoming of the standard RNN is the vanishing gradient problem, making it hard to transmit the gradient information consistently over long time. This is a crucial issue in designing a RNN model, and the long short-term memory (LSTM) was proposed specially to address this problem. The LSTM defines a new neuron or cell structure in the hidden layer with three additional multiplicative gates: the input gate, forget gate and output gate. These new cells are referred as memory cells, which allow the LSTM to learn meaningful long-range interdependencies. The structure of the memory cells is described in Fig. 5. σ is the logistic sigmoid function, realising the non-linearity of LSTM. The *cell* activation is a summation of the previous *cell* activation and the *input modulation*, which are controlled by the *forget gate* and *input gate* respectively. These two gates trade off the influences between the previous memory cell and current input information. The *output gate* controls how much of the *cell* activation to transfer to the final hidden state. Each LSTM hidden layer includes 128 LSTM memory cells, each of which has

a structure shown in Fig. 5.

[0031] The sequence labelling of varying lengths is processed by recurrently implementing the LSTM memory for each sequential input x^t (t is an integer from 1 to T), such that all LSTMs share the same parameters. The output of the LSTM h_t is used to feed to the LSTM at next input x_{t+1} . It is also used to compute the current output, which is transformed to the estimated probabilities over all possible characters. It finally generates a sequence of the estimations with the same length of input sequence, $P = \{p_1, p_2, p_3, \dots, p_\tau\}$.

[0032] Due to the unsegmented nature of the word image at the character level, the length of the LSTM outputs (T) is not consistent with the length of a target word string, $|S_w| = K$. This makes it difficult to train the RNN directly with the target strings. To this end, a connectionist temporal classification (CTC) is applied to approximately map the LSTM sequential output (p) into its target string as follow:

$$S_w^* \approx \mathbf{B}(\arg \max_{\pi} P(\pi | p)) \quad (5)$$

where the projection \mathbf{B} removes the repeated labels and the non-character labels. For example, $\mathbf{B}("-gg-o-oo-dd-")="good"$. The CTC looks for an approximately optimized path (π) with maximum probability through the LSTMs output sequence, which aligns the different lengths of LSTM sequence and the word string.

[0033] The CTC is specifically designed for the sequence labelling tasks where it is hard to pre-segment the input sequence to the segments that exactly match a target sequence. In our RNN model, the CTC layer is directly connected to the outputs of LSTMs, and works as the output layer of the whole RNN. It not only allows the model to avoid a number of complicated post-processing (e.g. transforming the LSTM output sequence into a word string), but also makes it possible to be trained in an end-to-end fashion by minimizing a overall loss function over $(X, S_w) \in \Omega$. The loss for each sample pair is computed as sum of the negative log likelihood of the true word string,

$$L(X, S_w) = -\sum_{i=1}^K \log P(S_w^i | X) \quad (6)$$

[0034] Finally, the RNNs model according to the present application follows a bidirectional LSTM architecture, as shown in Fig. 4. It has two separate LSTM hidden layers that process the input sequence forward and backward, respectively. Both hidden layers are connected to the same output layers, allowing it to access to both past and future information in the sequence.

[0035] In an example, the CNN model according to the present application is trained on about 1.8×10^5 character images and the CNN sequence is generated by applying the trained CNN with a sliding-window on the word images, followed by a column-wise normalization. The RNN model contains a bidirectional LSTM architecture. Each LSTM layer has 128 LSTM cell memory blocks. The input layer of our RNN model has 128 neurons (corresponding to the dimensions of the CNN sequence, $x_i \in R^{128}$), which are fully connected to both hidden layers. The outputs of two hidden layers are concatenated, and then fully connected to the output layer of the LSTM with 37 output classes (including an additional non-character class), by using a softmax function. Thus our RNN model has 273445 parameters in total, which are initialized with a Gaussian distribution of mean 0 and standard deviation 0.01 in the training process.

[0036] The recurrent model is trained with steepest descent. The parameters are updated per training sequence by using a learning rate of 10^{-4} and a momentum of 0.9. Each input sequence is randomly selected from the training set. A forward-backward algorithm is performed to jointly optimize the bidirectional LSTM and CTC parameters, where a forward propagation is implemented through whole network, followed by a forward-backward algorithm that aligns the ground truth word strings to the LSTM output maps, $\pi \in B^{-1}(S_w)$, $\pi' \rho \in R^{37 \times T}$. The loss function of E.q. (6) is computed approximately as,

$$L(X, S, \tau) \ll - \sum_{t=1}^{\tau} \log P(n_t | X) \quad (7)$$

[0037] Finally the approximated error is propagated backward to update the parameters. The RNN is trained on about 3000 word images, taken from the training sets of three benchmarks mentioned in the next section.

[0038] The performance of the proposed solution of text recognition is compared against start-of-the-art methods on three standard benchmarks for cropped word image recognition. The experiment result shows demonstrated that the method and apparatus of the present application has good performance in recognizing words in images with training by relative small number of samples.

[0039] Although the preferred embodiments of the present invention have been described, many modifications and changes may be possible once those skilled in the art get to know some basic inventive concepts. The appended claims are intended to be construed comprising these preferred embodiments and all the changes and modifications fallen within the scope of the present invention.

[0040] It will be apparent to those skilled in the art that various modifications and variations could be made to the present application without departing from the spirit and scope of the present invention. Thus, if any modifications and variations lie within the spirit and principle of the present application, the present invention is intended to include these modifications and variations.

Claims

1. A method for recognizing text in an image, comprising:
 - encoding the image into a first sequence with a convolutional neural network (CNN), wherein the first sequence is an output from a last second convolutional layer of the CNN;
 - decoding the first sequence with a recurrent neural network (RNN) into a second sequence, which has a same length as the first sequence and indicates estimated probabilities over all possible characters corresponding to each component in the first sequence; and
 - mapping the second sequence into a word string in which repeated labels and non-character labels are removed.

2. The method of claim 1, wherein encoding the image into a first sequence with a CNN comprises:
 - convoluting the image as a whole with the CNN, wherein a result of the convolution obtained from the last second convolutional layer of the CNN is the first sequence.

3. The method of claim 1, wherein encoding the image into a first sequence with a CNN comprises:
 - applying a sliding window to the image to divide the image into segments having a same size; and
 - convoluting the segments, individually and sequentially, with the CNN, wherein results of the convolution obtained from the last second convolutional layer of the CNN are components forming the first sequence.

4. The method of claim 1, prior to the step of encoding, further comprising, resizing the image to have a predetermined size.

5. The method of claim 4, wherein the CNN has been trained with image samples having the predetermined size and outputs 36 classes of different characters at a last layer of the CNN.

6. The method of claim 1, wherein the output from the last second convolutional layer of the CNN is just one neuron.

7. The method of claim 1, wherein the RNN has been trained with a set of convolutional sequence and corresponding word strings.

8. The method of claim 1, wherein the CNN comprises a maxout CNN.

9. The method of claim 1, wherein the RNN comprises a bidirectional long short-term memory (LSTM) layer and a connectionist temporal classification (CTC) layer, wherein the LSTM layer generates the second sequence, and the CTC layer generates the word string.

10. An apparatus for recognizing text in an image, comprising:
a convolutional neural network (CNN) encoding the image into a first sequence, wherein the first sequence is an output from a last second convolutional layer of the CNN; and

a recurrent neural network (RNN) decoding the first sequence into a second sequence, which has a same length as the first sequence and indicates estimated probabilities over all possible characters corresponding to each component in the first sequence;

wherein the RNN further maps the second sequence into a word string in which repeated labels and non-character labels are removed.

11. The apparatus of claim 10, wherein the CNN encodes the image into a first

sequence by:

convoluting the image as a whole with the CNN, wherein a result of the convolution obtained from the last second convolutional layer of the CNN is the first sequence.

12. The apparatus of claim 10, wherein the CNN encodes the image into a first sequence by:

applying a sliding window to the image to divide the image into segments having a same size; and

convoluting the segments, individually and sequentially, with the CNN, wherein results of the convolution obtained from the last second convolutional layer of the CNN are components forming the first sequence.

13. The apparatus of claim 10, wherein the image has been resized to have a predetermined size before being input to the CNN.

14. The apparatus of claim 13, wherein the CNN has been trained with image samples having the predetermined size and outputs 36 classes of different characters at a last layer of the CNN.

15. The apparatus of claim 10, wherein the output from the last second convolutional layer of the CNN is just one neuron.

16. The apparatus of claim 10, wherein the RNN has been trained with a set of convolutional sequence and corresponding word strings.

17. The apparatus of claim 10, wherein the CNN comprises a maxout CNN.

18. The apparatus of claim 10, wherein the RNN comprises a bidirectional long

short-term memory (LSTM) layer and a connectionist temporal classification (CTC) layer, wherein the LSTM layer generates the second sequence, and the CTC layer generates the word string.

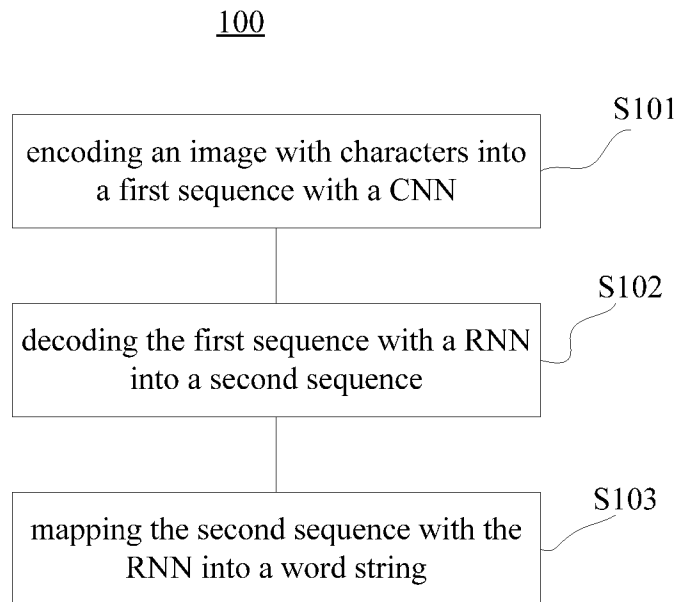


Figure 1

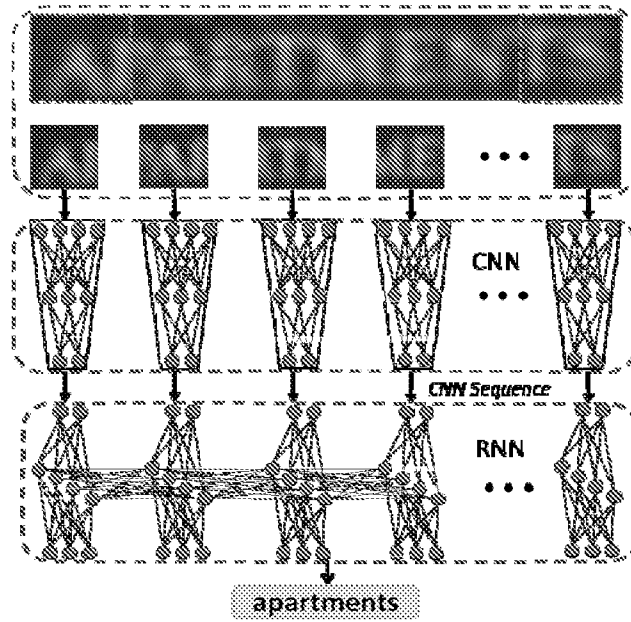


Figure 2

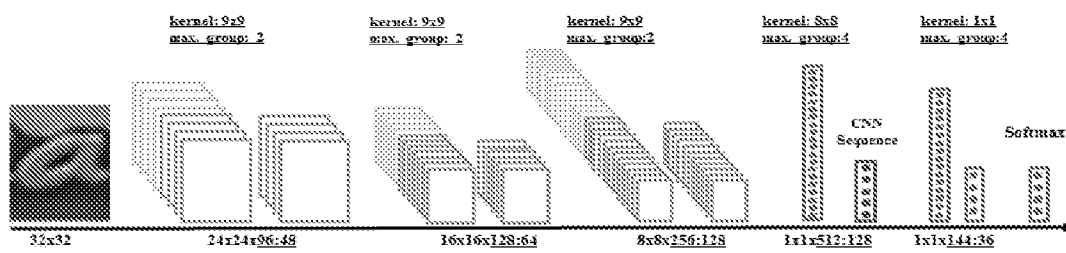


Figure 3

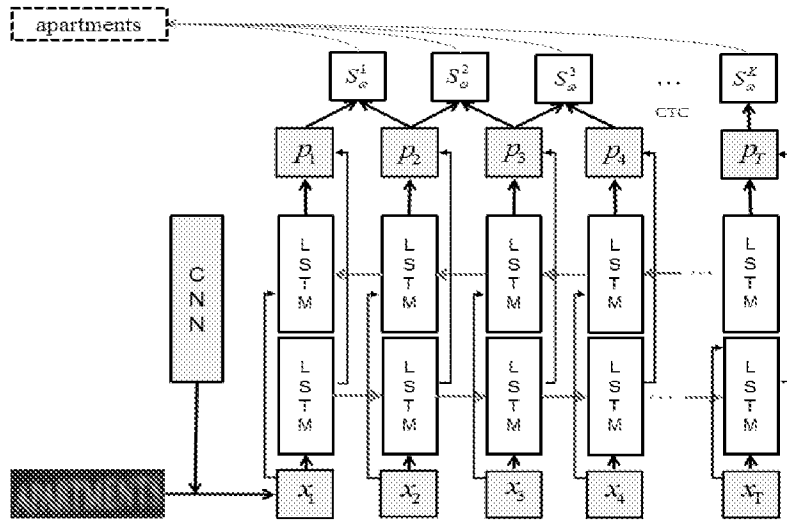


Figure 4

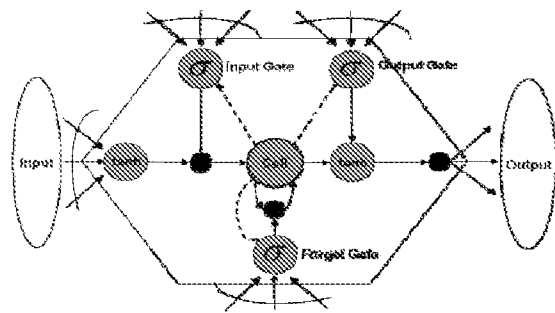


Figure 5

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2015/081308

A. CLASSIFICATION OF SUBJECT MATTER		
G06K 9/18(2006.01)i		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED		
Minimum documentation searched (classification system followed by classification symbols)		
G06K; G08G		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)		
WPI,EPODOC,CNPAT,CNKI: text, document, character, letter, recognize, identify, distinguish, image, picture, sequence, encode, code, decode, layer, neural, network, probability, label		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2006045341 A1 (SAMSUNG ELECTRONICS CO., LTD.) 02 March 2006 (2006-03-02) description, paragraphs [0032] and [0036]	1-18
A	KR 20130122842 A (KOREA INST. IND. TECHNOLOGY) 11 November 2013 (2013-1 1-1 1) the whole document	1-18
A	CN 1694130 A (UNIV. SHANGHAI ET AL.) 09 November 2005 (2005-1 1-09) the whole document	1-18
A	CN 101957920 A (UNIV. NAT. DEFENSE TECHNOLOGY PLA) 26 January 201 1 (201 1-01-26) the whole document	1-18
<p>I Further documents are listed in the continuation of Box C. <input checked="" type="checkbox"/> See patent family annex.</p>		
* Special categories of cited documents:		
“A”	document defining the general state of the art which is not considered to be of particular relevance	“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
“E”	earlier application or patent but published on or after the international filing date	“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
“L”	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
“O”	document referring to an oral disclosure, use, exhibition or other means	“&” document member of the same patent family
“P”	document published prior to the international filing date but later than the priority date claimed	
Date of the actual completion of the international search	Date of mailing of the international search report	
08 December 2015	29 February 2016	
Name and mailing address of the ISA/CN	Authorized officer	
STATE INTELLECTUAL PROPERTY OFFICE OF THE P.R.CHINA 6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing 100088, China	CI,Xue	
Facsimile No. (86-10)62019451	Telephone No. (86-10)62413237	

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2015/081308

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
US	2006045341	A1	02 March 2006	KR	20060020154	A	06 March 2006
KR	20130122842	A	11 November 2013	None			
CN	1694130	A	09 November 2005	None			
CN	101957920	A	26 January 2011	None			