

# 령역제안망을 리용한 필기수자렬인식의 한가지 방법

리진원, 김순돌

필기수자 및 필기수자렬인식은 필기문자인식분야에서 중요한 자리를 차지한다.

개별필기수자인식[2]에서는 CNN(중첩신경망)의 도입으로 하여 그 성능이 99.7%를 넘어서므로써 인간의 인식능력을 룡가하고있다. 그러나 필기수자렬인식[3]에서는 필기수자 분리문제의 어려움으로 하여 그것의 정확도개선에 한계를 가지고있다.

론문에서는 령역제안망(Region Proposal Network)을 리용하여 분리에 의존하지 않고 필기수자렬을 인식하는 한가지 방법을 제안하였다.

## 1. 문제 설정

필기수자렬화상은 필기한 수자들이 왼쪽으로부터 오른쪽으로 씌여진 화상이며 필기수자렬은 필기문서에서 생년월일과 같은 날자, 전화번호, 등록번호, 수값과 같은 수자렬의 형태로 나타난다.(그림 1)

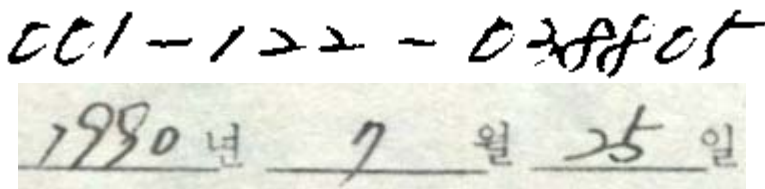


그림 1. 필기수자렬화상의 실례

고전적인 필기수자렬인식방법[1]들은 먼저 개별적인 필기수자령역을 분리한 다음 분리된 개별필기수자화상을 인식하는것이 대표적이다.

그러나 그림 1에서 보는것처럼 수자자체는 하나의 련결성분이지만 필기과정에 수자들이 불거나 필기도구나 종이의 질이 좋지 못하여 화상입력과정에 수자를 구성하는 획이 끊어져 1개의 수자가 여러개의 련결성분으로 되는 경우 정확한 개별수자령역을 확정하기 어렵다. 더우기 양식선우에 수자가 씌여진 경우 개별수자분리는 더 어려워진다. 이것은 분리에 기초한 필기수자렬인식방법으로는 개별수자인식의 정확도가 아무리 높다 하여도 전체적인 수자렬인식정확도를 높이는데서 한계가 있다는것을 의미한다.

이로부터 론문에서는 분리에 의존하지 않는 한가지 필기수자렬인식방법으로서 령역제안망을 리용하는 필기수자렬인식방법을 제안하였다.

## 2. 필기수자렬인식을 위한 령역제안망의 설계

### 1) 령역제안망에 의한 대상검출 및 인식원리

문자(수자)를 하나의 대상으로 보면 문자분리문제는 대상검출문제로 볼수 있으며 대상검출 및 인식방법을 문자렬인식에 적용할수 있다.

령역제안망[4]은 회귀에 의해 대상령역의 자리표를 예측하고 분류기에 의해 대상의 클래스를 결정하는 방법으로 대상검출 및 인식을 동시에 진행하는 중첩신경망이다.

일반적으로 분류대상모임을  $K = \{1, 2, \dots, k\}$ , 입력을  $\mathbf{x}$ , 모형을  $y = f(\mathbf{x})$  라고 표기할 때 분류(classification)문제는  $f: \mathbf{R}^n \rightarrow K$  로, 회귀(regression)문제는  $f: \mathbf{R}^n \rightarrow R$  로 표현된다. 여기서  $R$  는 실수모임이다.

분류문제는 입력대상이 어느 클래스에 속하는가를 결정하는것으로서 출력은 클래스 침수에 해당하는 옹근수로 되며 회귀문제는 입력대상이 주어졌을 때 1개의 수값을 예측하는것으로서 출력은 실수로 된다.

대상검출 및 인식에서 대상인식결과는 대상클래스의 침수로 얻어지기때문에 분류문제로 되며 대상검출결과는 대상령역4각형의 자리표(중심( $x, y$ ), 너비, 높이( $w, h$ ))의 4개 수값으로 얻어지기때문에 이 자리표들의 예측은 회귀문제로 된다.

본문에서는 대상령역자리표와 대상클래스결정을 모두 회귀에 의해 진행하는 령역제안망을 설계하였다.

회귀파라미터결정을 위한 학습자료는  $N$ 개의 자리표쌍  $\{P^i, G^i\}, i = \overline{1, N}$  으로 주어진다. 여기서

$$P^i = (P_x^i, P_y^i, P_w^i, P_h^i), G^i = (G_x^i, G_y^i, G_w^i, G_h^i)$$

는 각각 제안4각형  $P^i$  와 정답4각형(Ground-truth)  $G^i$  의 자리표 즉 4각형의 중심과 너비, 높이를 표현한다. 앞으로 침수  $i$  는 생략하고 설명한다.

정답4각형은 학습화상속에 들어있는 표식화된 대상령역4각형들이며 제안4각형은 CNN의 마지막출력특징지도의 매 화소에 대응하는 원화상위치(살창 또는 세포)에서 설정하는 미리 정의된 너비, 높이를 가지는 4각형들이다. 여기서 제안이라는 의미는 원화상의 매 살창점에서 여러가지 너비, 높이를 가지는 4각형들을 제출하고 그 4각형들이 대상을 포함하는가를 판정해본다는것이다.

자리표회귀를 위한 학습의 목표는 제안4각형  $P$  를 정답4각형  $G$  로 넘기는 변환함수를 학습하는것이다. 변환은 제안4각형  $P$  에 대한 4개의 함수  $d_x(P), d_y(P), d_w(P), d_h(P)$  로 실현된다. 여기서  $d_x(P), d_y(P)$  는  $P$  의 중심에 대한 척도불변변환이며  $d_w(P), d_h(P)$  는  $P$  의 너비, 높이에 대한 로그공간변환이다.

제안  $P$  로부터 정답예측  $\hat{G}$  에로의 변환은 식 (1)–(4)에 의해 진행된다.

$$\hat{G}_x = P_w d_x(P) + P_x \quad (1)$$

$$\hat{G}_y = P_h d_y(P) + P_y \quad (2)$$

$$\hat{G}_w = P_w \exp(d_w(P)) \quad (3)$$

$$\hat{G}_h = P_h \exp(d_h(P)) \quad (4)$$

함수  $d_\Delta(P)$  (여기서  $\Delta$  는  $x, y, w, h$  중의 하나)는 CNN의 마지막중첩층의 특징  $\phi(P)$  (특징지도를 벡토르화한것.)에 대한 선형함수로 모형화한다.

$$d_\Delta(P) = \mathbf{w}_\Delta^T \phi(P) \quad (5)$$

$\mathbf{w}_\Delta$  는 선형모형의 무게파라미터로서 다음의 최소2제곱목적함수

$$\mathbf{w}_\Delta = \arg \min_{\hat{\mathbf{w}}_\Delta} \sum_i^N (t_\Delta^i - \hat{\mathbf{w}}_\Delta^T \phi(P^i))^2 + \lambda \|\hat{\mathbf{w}}_\Delta\|^2 \quad (6)$$

에 의하여 얻는다.

학습쌍  $(P, G)$  에 대한 회귀목표  $t_\Delta$  는 다음과 같이 정의된다.

$$t_x = (G_x - P_x) / P_w \quad (7)$$

$$t_y = (G_y - P_y) / P_h \quad (8)$$

$$t_w = \log(G_w / P_w) \quad (9)$$

$$t_h = \log(G_h / P_h) \quad (10)$$

식 (6)의 조절파라미터  $\lambda$  는  $\lambda = 1\,000$  으로 설정된다.

한편 학습쌍  $(P, G)$  들중에서 서로 겹치지 않는 쌍  $(P, G)$  는 의미가 없기때문에 학습에서 제외된다.

두 4각형의 겹침 IoU(Intersection over Union)는 다음과 같이 계산된다.

$$IoU(P, G) = \frac{|P \cap G|}{|P \cup G|} \quad (11)$$

여기서 분자는 두 4각형의 사림면적이고 분모는 두 4각형의 합면적을 의미하며 다음과 같이 계산된다.

겹침부분4각형  $O$  의 너비, 높이  $(O_w, O_h)$  는

$$O_w = \min(P_x + P_w, G_x + G_w) - \max(P_x, G_x)$$

$$O_h = \min(P_y + P_h, G_y + G_h) - \max(P_y, G_y)$$

로 계산되며 이때 사림면적은

$$|P \cap G| = \begin{cases} O_w \cdot O_h, & O_w > 0 \text{ 이고 } O_h > 0 \\ 0, & O_w \leq 0 \text{ 또는 } O_h \leq 0 \end{cases} \quad (12)$$

이고 합면적은

$$|P \cup G| = (P_w P_h + G_w G_h) - |P \cap G| \quad (13)$$

로 된다.

논문에서는  $IoU > 0.5$  일 때 두 4각형이 겹친것으로 설정하였다.

대상인식을 위한 회귀도 유사한 방법으로 계산된다. 즉 매  $k \in K$  에 대하여 변환

$$d_k(P) = \mathbf{w}_k^T \phi(P) \quad (14)$$

와 목적함수

$$\mathbf{w}_k = \arg \min_{\hat{\mathbf{w}}_k} \sum_i^N (1 - \hat{\mathbf{w}}_k^T \phi(P^i))^2 + \lambda \|\hat{\mathbf{w}}_k\|^2 \quad (15)$$

에 의해 대상클래스회귀가 진행된다.

대상회귀결과는 해당 세포가 정답클래스를 포함하고있을 때 1에 가까운 값을, 아닐 때 0에 가까운 값을 가진다. 결국 1개 제안에 대해  $(4+K)$ 개의 회귀가 진행된다.

## 2) 필기수자렬인식을 위한 영역제안망의 구조설계

필기수자렬인식을 위한 영역제안망은 특징추출층, 영역제안층, 수자검출 및 인식층의 세가지 부분체계로 구성된다.(그림 2)

특징추출층은 일반중첩신경망과 마찬가지로 입력화상으로부터 여러개의 중첩층을 거쳐 마지막중첩층의 특징지도를 출력으로 한다. 영역제안층은 특징추출층으로부터 출력된 특징지도를 입력으로 하고 특징지도의 매 화소에서 여러개의 고정4각형을 제안하여 매

제안4각형에 대한 특징을 출력한다.

수자검출 및 인식층은 매 제안4각형특징에 대한 회귀값들로부터 수자코드와 수자령역자리표를 계산하여 출력한다.

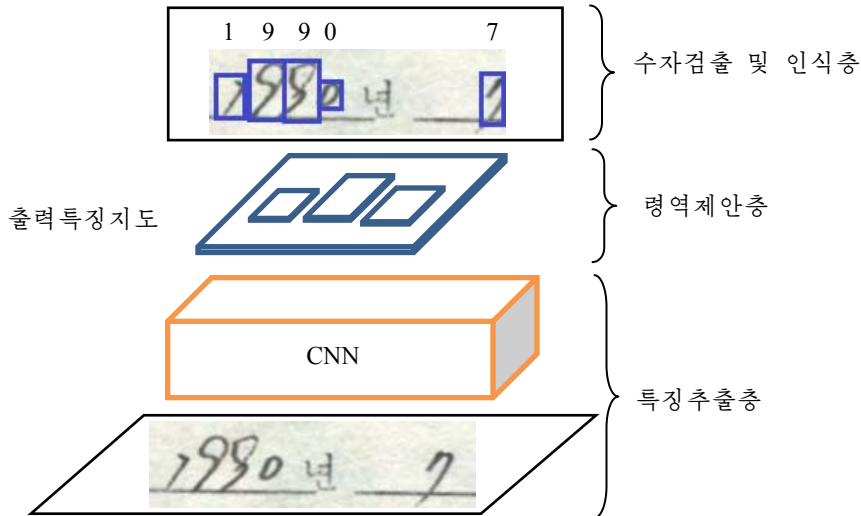


그림 2. 필기수자열인식을 위한 지역제안망의 구조

출력특징지도의 크기는 대상검출의 정밀도와 관계되며 일정한 한계를 가진다.

출력특징지도의 크기가 클수록 더 작은 대상(수자)을 검출할수 있지만 특징지도의 해상도를 1/2로 축소하는  $2 \times 2$  최대저장층(MaxPooling)의 개수가 고정되어있을 때 특징지도의 크기를 크게 하자면 입력정규화의 크기가 커져야 한다.

결국 출력특징지도의 크기는 입력정규화화상의 크기와 최대저장층의 개수에 따라 제한을 가진다.

논문에서는 필기수자열화상인 경우 우아래로는 1개의 수자만을 검출하면 되고 좌우로는 최대 20정도의 수자들을 검출할수 있어야 한다는것을 고려하여 출력특징지도의 크기를  $38 \times 7$ 로 정하였다.

최대저장층의 개수가 4개인 경우 입력정규화화상의 크기는

$$(38 \times 2^4) \times (7 \times 2^4) = 608 \times 112$$

로 된다. 이때 출력특징지도화상의 어떤 화소의 위치  $(x_0, y_0)$ 은 입력정규화화상의 위치

$$(x, y) = (x_0 \cdot 2^4, y_0 \cdot 2^4) = (16x_0, 16y_0)$$

에 대응된다.

결국 출력특징지도의 매 화소에서 지역제안을 진행한다는것은 입력정규화화상에서 간격 16으로 부분창문을 주사식으로 이동하면서 지역제안을 진행한다는것을 의미하며 출력특징지도의 크기가  $38 \times 7$ 일 때  $38 \times 7 = 266$ 개의 부분창문에서 지역제안을 진행한다는것을 의미한다.

지역제안층은 물리적으로 존재하는 층은 아니지만 망의 구성에서 출력특징지도에 여러개의 고정4각형을 제안하여 매 제안4각형에 대한 특징을 얻는다는 의미에서 하나의 층으로 고찰한다.

지역제안층에서 중요한 문제는 고정4각형의 결정문제이다.

논문에서 검출하려는 대상은 필기수자로서 보통 너비-높이비가 1보다 작은 세로장 방형4각형영역을 차지하므로 고정4각형의 너비-높이비는  $\{0.2, 0.5, 0.7, 1.0\}$ 의 4가지로 정한다. 또한 입력정규화화상의 높이가 112인것을 참고하여 척도는  $\{30, 50, 80\}$ 의 세가지로 정한다.

결국 전체 고정4각형의 개수는  $4 \times 3 = 12$ 개이다.

수자검출 및 인식층은  $1 \times 1$  려파기조를 가지고있는 하나의 중첩층이다.(그림 3)

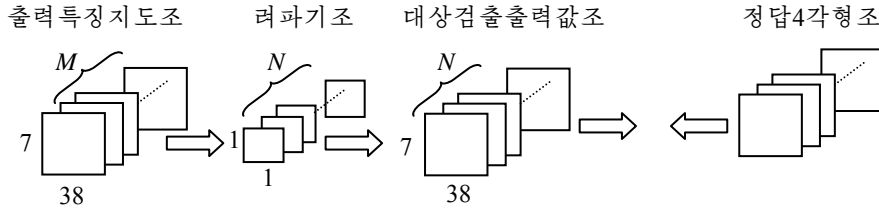


그림 3. 수자검출 및 인식층의 구조

입력은 특징추출층의  $M$ 개의 출력특징지도조이며  $N$ 개의  $1 \times 1$  려파기를 통과하여  $N$ 개의 대상검출출력값조를 출력한다. 여기서  $1 \times 1$  려파기개수  $N$ 은 고정4각형개수가  $N_a$ , 수자클래스개수가  $K$  일 때

$$N = N_a \times (4 + K) \quad (16)$$

로서 논문에서는

$$N = N_a \times (4 + K) = 12 \times (4 + 10) = 168$$

이다. 이것은 1개의 출력특징점당  $N = 168$  개의 예측이 진행된다는것을 의미한다.

필기수자열인식을 위한 영역제안망의 매 층의 파라미터구성을 표 1에 보여주었다.

표 1. 영역제안망의 매 층의 파라미터구성

중첩층	입력특징지도			려파기조				출력특징지도			계산량/ 적합개수 ( $10^9$ )
	너비	높이	통로수	핵의 너비	핵의 높이	려파기 개수	파라메터 개수	너비	높이	통로수	
conv1/mp2	608	112	3	3	3	16	432	304	56	16	0.06
conv2/mp2	304	56	16	3	3	32	4 608	152	28	32	0.16
conv3/mp2	152	28	32	3	3	64	18 432	76	14	64	0.16
conv4/mp2	76	14	64	3	3	128	73 728	38	7	128	0.16
conv5	38	7	128	3	3	512	589 824	38	7	512	0.31
conv6	38	7	512	3	3	1 024	4 718 592	38	7	1 024	2.51
conv7	38	7	1 024	1	1	168	172 032	38	7	168	0.09

### 3. 실험 결과

제안방법의 유효성을 판정하기 위하여 각종 필기양식문서에 찍여진 날자부분화상 5 000개로 구성된 화상모임을 준비하였다. 이 화상모임을 9:1의 비율로 나누어 학습 및 검사자료모임으로 하였다. 학습은 GTX-980이 장비된 Corei7 3.6GHz컴퓨터와 tensorflow호

레이워크상에서 진행되었다.

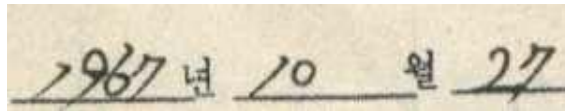
4만회의 학습을 진행하여 제안방법과 선행방법[1]을 비교하였다.(표 2)

표 2는 제안방법이 분리에 의존한 선행방법[1]에 비해 인식시간을 줄이면서도 인식 정확도를 5%이상 제고함으로써 필기수자열인식에서 대폭적인 성능개선을 달성하였다는 것을 보여주고있다.

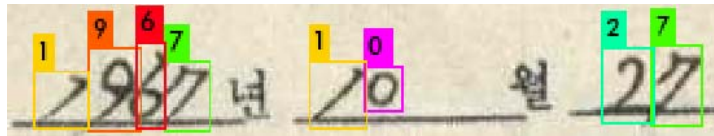
표 2. 성능평가

	화상모임	화상수/개	오유수/개	인식률/%	평균인식시간/ms
선행방법[1]	학습	4 500	236	94.75	150
	검사	500	36	92.8	
제안방법	학습	4 500	32	99.28	120
	검사	500	14	97.2	

그림 4의 ㄱ)에서 보여준 입력화상은 필기수자들이 서로 불거나 양식선과 사귀고있기때문에 고전적인 수법으로는 정확한 결과를 얻기 어렵다. 그러나 ㄴ)에서 보는것처럼 제안방법에 의하여 정확히 인식되고있다는것을 알수 있다.



ㄱ)



ㄴ)

그림 4. 필기수자열인식결과의 실례

ㄱ) 입력화상, ㄴ) 제안방법의 인식결과

## 맺 는 말

분리에 의존하지 않는 한가지 필기수자열인식방법으로서 영역제안망을 리용하는 필기수자열인식방법을 제안하고 실험을 통하여 그 유효성을 확증하였다.

## 참 고 문 헌

[1] 김일성종합대학학보(자연과학), 52, 4, 37, 주제95(2006).

[2] X. -X. Niu, C. Y. Suen; Pattern Recognition, 45, 6, 1318, 2012.

[3] A. G. Hochuli et al.; Pattern Recognition, 78, 6, 1, 2018.

[4] S. Ren et al.; IEEE Transactions Pattern Analysis and Machine Intelligence, 39, 6, 1137, 2017.

주제110(2021)년 2월 5일 원고접수

## **A Method of Handwritten Digits Recognition Using Region Proposal Network**

*Ri Jin Won, Kim Sun Dol*

In this paper we designed a segmentation-free handwritten digits recognition system using region proposal network and confirmed that the proposed method achieved high recognition accuracy over other preceding methods by experiments.

Keywords: region proposal network, handwritten digits recognition, deep neural network