

論文99-36C-4-5

동적 프로그래밍 정합을 이용한 효율적인 필기 단어 인식 방법

(An Approach for Efficient Handwritten Word Recognition Using Dynamic Programming Matching)

金敬煥 *

(Gyeonghwan Kim)

요 약

본 논문에서는 실제 응용분야에서 사용될 수 있는 효율적인 필기 영어 단어 인식 방법을 제안한다. 필기 단어인식과 관련된 대부분의 응용분야에서 제공되는 사전의 활용을 극대화하기 위해 사전단어들을 인식의 초기 단계에서부터 사용한다. 초과 분할된 단어의 세그먼트들과 사전단어들 사이의 정합을 위해 동적 프로그래밍을 사용하며, 정합구간을 가변적으로 조정할 수 있도록 학습단계에서 추출한 문자 분할과 관련된 통계를 활용한다. 또한, 사전단어의 각 문자와 세그먼트들 사이의 정합 결과를 저장하여 반복되는 계산을 피한다. 제안하는 방법의 효율성을 입증하기 위해 다양한 서체를 갖는 실험용 필기 단어영상을 사용하여 실험을 수행한 결과, 사전에 기반한 단어 인식 과정을 최대로 활용하기 위한 가변정합구간 개념 및 문자단위 정합결과 저장 방법이 동적 프로그래밍과 함께 인식 속도 및 정확도 향상에 모두 크게 기여함을 확인하였다.

Abstract

This paper proposes an efficient handwritten English word recognition scheme which can be applied practical applications. To effectively use the lexicon which is available in most handwriting related applications, the lexicon entries are introduced in the early stage of the recognition. Dynamic programming is used for matching between over-segmented character segments and letters in the lexicon entries. Character segmentation statistics which can be obtained while the training is being performed are used to adjust the matching window size. Also, the matching results between the character segments and the letters in the lexicon entries are cached to avoid repeat of the same computation. In order to verify the effectiveness of the proposed methods, several experiments were performed using thousands of word images with various writing styles. The results show that the proposed methods significantly improve the matching speed as well as the accuracy.

I. 서 론

컴퓨터 기술의 발달에 따라 다양한 정보 전달 및 저장 매체와 관련 기술이 함께 발달하고 있으나, 가장 전통적인 매체인 필기도구와 종이는 계속 중요하게 사용되고 있다. 이에 따라 종이에 쓰여진 필기서술 자체

를 컴퓨터를 사용해 인식하려는 노력이 지난 40여 년 동안 계속되어 왔으나 우편물 분류^[1], 개인수표 필기 금액 인식^[2], 세금계산서 또는 보고서의 금액 및 기재사항 인식^[3] 등의 응용분야에서 제한적으로 활용되고 있는 상황이다.

필기서술 인식의 어려움은 인식 대상의 모양의 다양성에서 출발한다. 사람마다 각각 다른 필기습관을 갖고 있으며, 종이 및 펜 등 필기도구의 차이, 데이터 취득과정에서 도입될 수 있는 잡음 및 왜곡현상 등을 주요 원인으로 분류할 수 있다. 경우에 따라서는 동일

* 正會員, 西江大學校 電子工學科
(Dept. of Electronic Engineering, Sogang University)
接受日字:1999年1月23日, 수정완료일:1999年3月29日

한 사람에 의해 쓰여진 필기서술이, 필기될 당시의 자세 및 종이의 여백 등과 같은 주변의 상황에 따라 영향을 받기도 한다. 특정 문자의 진후에 위치하는 문자에 따라 모양이 다르게 쓰여지는 현상도, 음성 신호해석에서 연음현상처럼, 자주 발견할 수 있다.

따라서, 한 단어를 구성하는 각각의 문자를 인식하고 그 결과를 종합하여 단어를 인식하는 방법은 인쇄된 문서의 해석에는 쉽게 적용될 수 있으나 필기 서술에서는 제한적으로 사용될 수밖에 없다. 이 문제를 완화시키기 위해 인식 후보 단어들을 포함하고 있는 사전을 작성하여 활용하는 방법이 일반적으로 사용되고 있다^[4].

본 논문은, 이처럼 사전을 사용하는 필기 영어단어 인식과 관련된 응용분야에서 인식의 정확도와 처리속도를 현실적으로 개선할 수 있는 방법을 제안한다. 문제를 명확하게 하기 위해 본 논문에서는 다음과 같은, 일반적으로 널리 사용되는, 가정을 전제로 한다: “인식 대상 영상은 독립된 한 단어를 포함하며 이 영상에 해당되는 사전이 공급된다.” 따라서 인식의 문제는 “입력 영상을 처리하여 주어진 사전에 포함된 단어들로부터 유사도가 가장 가까운 단어를 선택한다”로 정의될 수 있다. 인식결과를 분석을 통해 가장 높은 유사도를 나타낸 단어의 취사선택 여부를 결정하여 인식시스템 전체의 신뢰도를 향상시킨다.

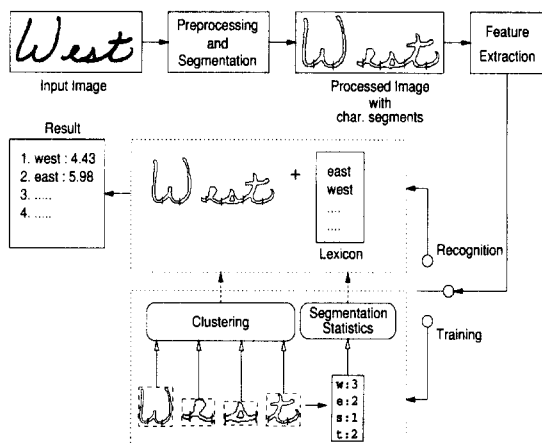


그림 1. 제안하는 단어 인식시스템의 구성
Fig. 1. Block diagram of the word recognition system.

위에서 주어진 전제조건을 인식단계에서 최대한 활용할 수 있도록, 동적 프로그래밍을 사용한다. 동적 프

로그래밍을 이용한 정합은 이미 오래 전부터 음성인식 등에 활용된 기법으로^[5], 본 논문에서도 그 장점들을 활용할 수 있도록 적용된다. 단어의 학습과정에서, 단어를 구성문자 또는 그 이하의 단위로 분할하는 과정에서 얻을 수 있는 분할과 관련된 통계를 인식과정에서 적용하며, 주어진 사전에 존재하는 단어들과의 정합과정에서 반복적인 연산과정을 피함으로서 제안하는 인식시스템의 정확도와 처리 속도를 함께 향상시킨다. 그림1에 본 논문에서 제안하는 단어 인식시스템의 전체적인 구성을 나타내었다.

II . 필기 단어인식의 방법 및 문제들

본 논문에서 제안하는 방법의 배경 및 동기를 설명하기 위하여 필기 단어 인식과 관련된 일반적인 사항들을 간단하게 살펴본다.

1. 문자분리(character segmentation)

필기 단어의 인식과 관련된 연구는, 제1절에서 언급된 가정들을 전제로 하더라도, 단어를 구성하는 문자들로 분리하는 문제와 밀접하게 연결되어 있고 결국 사전의 활용방법과 연결된다. 필기단어는 구성문자의 연결정도에 따라 크게 cursive, discrete, 그리고 두 형태의 혼합(mixed)으로 구분될 수 있다. 필기자에게 우편번호를 기입하는 칸처럼 미리 인쇄된 상자들을 제공하는 등의 상당한 제한을 두기 전에는, 문자분리 문제가 없는 필기형태를 기대하기는 어렵다. Cursive 또는 mixed 형태의 필기에 있어서 구성 문자간 경계는 일반적으로 명확하지 않으며 또한 복잡한 문자는 분할하기에 따라 다양한 문자의 조합으로 해석할 수 있기 때문에 별도의 인식과정의 도움 없이 문자경계를 분리하기는 거의 불가능하다. 예를 들어 'm' 또는 'w'와 같은 문자들은 분리하기에 따라 두 개의 'n' 또는 'i'와 'u'등으로 해석 가능하다. 결국 단어를 구성하는 문자들로의 분리문제는, 단어를 인식하기 위해 분리하지만 그 명확한 분리를 위해 인식이 필요하다는 유명한 역설^[6]에 직면하게 된다.

단어영상의 분할과 관련된 문제를 완화하기 위해 제안된 방법이 holistic 방식이며 이 경우에는 단어의 전체적인 모양으로부터 추출한 특징들을 이용하여 단어를 인식한다^[7].

따라서, 단어를 인식기의 도움 없이 단어영상으로부터 추출 가능한 특징들을 활용하여 분할하게되고, 이

경우 문자 사이의 경계를 놓치지 않기 위해 단어를 구성하는 문자의 수 보다 많은 수의 문자조각을 구하는 초과분할방법을 피할 수 없게 된다. 결국 초과 분할된 단어를 인식단계에서 의미 있는 단어로 해석하기 위해 사전의 도움이 필요하게된다.

2. 사전의 활용 방식

상당수의 필기단어 인식시스템에서 입력 단어영상의 후보 단어들의 집합인 사전은 어렵지 않게 제공될 수 있다. 예를 들어, 우편물의 주소를 해석하는 경우에는 우편번호가 인식되면 그 우편번호와 연관된 도시이름들을 데이터베이스에서 추출하여 사전으로 사용할 수 있고^[1], 미국 및 서구에서 많이 사용되는 개인수표의 금액 인식의 경우에는 금액을 서술하는데 사용되는 단어들의 집합을 사전으로 사용할 수 있다.

사전의 활용은 인식방법에 따라 구분 될 수 있으나, 근본적으로 위에서 언급한 것처럼 단어를 구성문자로 초과 분할함에 따라 있을 수 있는 해석상의 애매모호성을 완화시킨다. 초과 분할된 문자조각의 각각 또는 하나 이상의 연속 조합을 문자인식을 통해 인식하는 경우에는 의미 있는 단어로의 해석을 위한 후처리 과정에 사전이 활용된다. 앞에서 언급한 holistic 방식의 경우에 사전은 각 후보 단어로부터 정합을 위한 관련 특징들을 제공하는 역할을 하게된다^[7].

3. 인식의 단위

필기단어의 인식문제는, 개별 문자인식의 연장으로 단어인식을 해석하는 방법(문자 중심의 단어인식 방법)과, 단어 자체를 인식 대상의 기본 단위로 보고 접근(단어 중심의 단어 인식방법)하는 두 가지 방법으로 분류될 수 있다. 전자의 경우, 문자 또는 초과분할에 따른 문자조각의 조합이 인식의 대상이 되며, 인식된 결과들을 의미 있는 단어로 해석하기 위해 후처리과정이 필요하게 된다. 반면 후자의 경우, 단어 자체를 인식의 대상으로 하며 정합과정에 사전이 활용되어 별도의 후처리 과정을 피할 수 있다.

그림2에 문자 중심 단어인식 방법과 단어 중심 단어인식 방법을 문자 세그먼트 또는 그 조합의 정합방법과 사전의 역할을 중심으로 나타내었다. 입력 단어 영상은 초과 분할되기 때문에 실제적으로는 많은 수의 문자 세그먼트를 조합하면서 문자 단위 정합을 수행하게되는데 그림2에서는 문자의 경계가 바르게 찾아졌을 경우에 대해서만 한 예로 보여주고 있다. 그림2(a)의

문자 중심 단어 인식방법의 경우, 일반적인 문자인식 방법을 각 문자 세그먼트 또는 가능한 세그먼트 조합에 대해 적용하여 문자 인식 결과를 모든 문자 class에 대해 구한다. 구한 결과를 해석하여 의미 있는 단어를 만드는 후처리 과정에서 사전이 활용되며, 사전에 있는 단어를 구성하기 위한 비용과 문자인식 결과를 조합하여 최종 단어 인식결과를 계산한다. 반면, 그림2(b)의 단어 중심 단어인식 방법의 경우, 문자 세그먼트 또는 그 조합과 비교중인 사전단어내의 문자들과의 정합이 수행되며, 이를 위해 사전이 문자 정합단계부터 관여한다. 현재 정합중인 사전단어에 포함된 문자의 cluster들과 문자세그먼트와의 정합만이 수행되고 그 결과를 종합하여 단어인식 결과를 직접 얻을 수 있기 때문에 별도의 후처리 과정이 필요하지 않게 된다.

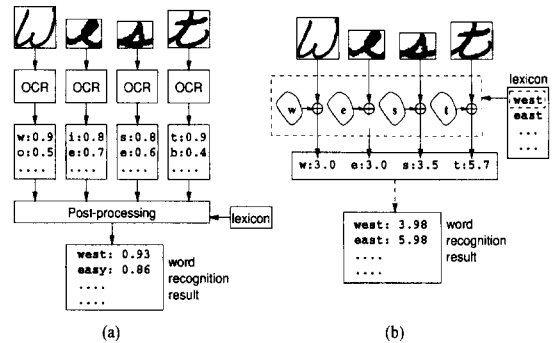


그림 2. 필기단어 인식방법의 비교: (a)문자 중심 및 (b) 단어중심 인식방법

Fig. 2. Comparison of handwritten word recognition schemes: (a) character based and (b) word based.

4. 제안하는 방법

본 논문에서는 인식의 정확도와 정합시간 향상의 관점에서, 초과문자분할을 이용하고 사전을 정합과정의 초기에 활용하는 ‘단어 중심의 단어 인식 방법’을 제안한다. 제1절에서 언급한 일반적인 제한사항과 이 절에서 제안하는 사항들을 효율적으로 활용하기 위한 정합수단으로 동적 프로그래밍을 사용한다.

Ⅲ. 동적 프로그래밍

동적 프로그래밍은 순차적인 결정이 필요한 분야에서 최적의 경우를 찾아야하는 문제를 해결하기 위해 광범위하게 연구되고 사용되어왔다. 음성과 같은 패턴

정합 문제에 있어서 연속적인 특징들을 비교해야 하는데, 동일한 발음에 대해서도 발음속도 등의 변화와 발음방법에 따라 순차적인 비교가 쉽지 않다. 따라서, 인식을 위해서는 발음속도의 변화와 특징들 사이의 간격 변화가 인식의 결과에 영향을 주지 않도록 해야 할 필요가 있다. 이를 위해 시간에 대해 특징을 재배열하고 정규화 할 필요가 있는데, 이를 동적 프로그래밍에 적용하여 정리한 기법이 dynamic time-warping이다. 이와 관련된 자세한 설명 및 응용 예들이 Rabiner와 Juang의 저서^[8]에 자세히 소개되어 있다.

동적 프로그래밍에 기초한 기법들은, 온라인 필기문자에서는 물론, 오프라인 필기문자 또는 단어에서 발견할 수 있는 여러 유사성들로 인해 다양한 형태로 적용되고 있다^[6,9]. 동적 프로그래밍 뿐 아니라 음성신호의 해석 또는 인식과 관련되어 개발된 Hidden Markov Models (HMMs)와 Time Delay Neural Network(TDNN)등의 기법을 적용한 노력이 대표적인 경우이다^[10,11]. 음성 신호에서 시간에 따른 특징의 변화를 필기서술에서는 쓰여진 순서(일반적으로 왼쪽에서 오른쪽으로)로 해석하고, 입력영상에 하나의 단어만 포함되었다고 가정하여 위에서 언급한 제한 사항을 독립된 단어영상의 해석에 적용할 수 있게 된다.

IV. 필기단어인식 시스템구현

실제 시스템 설계 및 구현에 있어서 중요하게 고려된 사항을 정리하면, 제1절에서 언급한 것처럼, 1) 대부분의 응용분야에서 인식 대상 단어영상과 관련된 사전은 제공되고, 2) 입력영상에는 하나의 단어만을 포함한다는 일반적인 조건들과, 이를 동적 프로그래밍 방법에 적용하기 위한 추가적인 해석으로서 1) 입력 단어영상은 초과분할방법에 따라 인식기능의 관여 없이 문자 또는 그 이하의 단위로 분리되고, 2) 영상은 분리된 독립단어를 포함하기 때문에 영상의 좌측에 처음 나타나는 필기 획을 단어영상의 시작으로, 우측에서 끝나는 필기 획을 단어영상의 마지막으로 간주하며, 3) 패턴정합을 위한 학습 중에 추가적으로 구할 수 있는 단어의 문자분할과 관련된 통계를 인식단계에서 효율적인 정합을 위해 활용한다. 제안하는 인식시스템은 이상의 단서들을 동적 프로그래밍으로 종합하여 입력 필기 단어영상과 가장 가까운 유사도를 갖는 단어를 주어진 사전으로부터 선택한다.

1. 학습

정합과정에 필요한 기준 특징벡터들을 제공하기 위해, 문자단위의 군집화 방법을 사용하여 코드북을 작성한다. 코드북의 code words는 21,054개의 문자영상을 사용하여 학습되었고 이 때 사용된 필기영상은 우편물의 주소로부터 수집한 미국의 도시 이름들을 포함하며 제5절에서 성능평가를 위해 사용되는 단어영상들과는 분리된 집단의 단어영상들이다. 군집화를 위해 LBG 알고리즘을 사용하였고, 각 문자 당 군집의 수는 인식시스템의 속도와 정확도를 고려하여, 오차정도와 허용 가능한 최대 군집의 수로 제한하여 조정하였다.

군집화 과정을 수행한 후, 모든 문자 class에 대한 모든 code words가 구해지며 이를 다음 식(1)과 같이 표기한다.

$$\mathbf{K}_m = \{ \mathbf{k}_m(i) \}_{i=1}^{K_N} \quad (1)$$

여기서, $m=a, b, \dots, z$ 이고, K_N 은 문자 m 에 대한 code words의 수이며 $\mathbf{k}_m(i)$ 는 문자 m 의 i 번째 code word를 의미한다.

문자 단위로 정합을 수행한 후, 동적 프로그래밍을 이용하여 단어를 인식하기 때문에, 학습단계에서 사용되는 문자의 모양과 인식단계에서 사용되는 문자는 가능한 비슷한 양상을 갖도록 전체시스템을 설계한다. 필기 단어영상에서 구성문자를 분할해 낼 때 분할점은 다양하게 설정될 수 있으며 이 설정에 따라 분할된 문자 영상의 모습은 같은 문자에 대해서도 차이가 날 수 있다. 따라서 인식시스템에 포함되어 있는 동일한 단어분할 방법을 학습용 단어영상에 적용하여 문자로 분할하여 학습용 문자영상으로 사용함으로써 실제 인식 과정에서 사용될 문자의 모습과 차이를 가능한 줄인다.

학습과정 중, 인식과정에서 사용되는 동일한 문자 분할 과정을 적용하는 또 다른 중요한 이유는, 각 문자를 일반적으로 어떤 양상으로 분할하는지에 대한 통계적인 자료를 구해 인식단계에서 활용하기 위함이다. 이 통계적인 자료는 동적 정합과 관련하여 가변정합구간(variable duration)으로 정의되어 사용되며, 이후에 소개되는 단어 인식과정에서 인식의 속도와 정확도를 모두 향상시키는 중요한 정보로 사용된다.

2. 각 문자의 가변정합구간

단어를 인식하는 과정에서, 구성문자의 정확한 경계를 파악할 수 없기 때문에 일반적으로 초과분할방법을

사용한다는 사실은 이미 앞에서 설명하였다. 따라서, 인식과정에서 초과 분할된 문자를 조합하며(즉, 문자단위의 인식을 수행하며) 가장 좋은 문자정합 결과를 내는 경우를 찾게된다. 문자분할 후, 입력 단어영상을 식(2)와 같이 표현한다.

$$S = \{s_0, s_1, \dots, s_{S_N-1}\} \quad (2)$$

여기서, S_N 는 단어분할 결과 얻어진 세그먼트의 수이며 s_i 는 i 번째 세그먼트를 의미한다. 또한, 문자 정합과정에서 필요한 초과 분할된 문자 세그먼트들의 조합을 식(3)과 같이 표현한다.

$$S(b, e) = (s_k)_{k=b}^e \quad 0 \leq b < S_N, 0 \leq e < S_N, b \leq e \quad (3)$$

여기서, b 는 시작 세그먼트 번호이고 e 는 끝나는 세그먼트 번호이다. 예를 들어 $S(3,5)$ 는 3번 세그먼트에서 5번 세그먼트까지 3개의 세그먼트를 조합한 결과를 나타내며 $S(4,4)$ 는 4번 세그먼트만을 의미한다.

가변정합구간이라는 개념의 도입은 단어인식과정을 최적화 할 수 있는 방법을 제공하기 위함이다. 학습과정에서 구할 수 있는 문자분할 통계는 일반적으로 각 문자들이 문자분할 알고리즘에 의해 어떠한 양상으로 분할되는 지를 나타낸다. 즉, 학습단계에서 수집된 특정 문자 class의 학습용 문자영상들이 일반적으로 몇 개의 문자 세그먼트로 구성되었는지에 대한 통계를 제공한다. 예를 들어, 간단한 모양을 한 'i' 또는 'l'과 같은 문자들은 이보다 복잡한 모양을 한 'w'와 'm'등의 문자들에 비해 적은 수의 세그먼트로 구성될 것이라는 사실을 쉽게 알 수 있다. 이러한 통계적 자료를 전체 사용 문자에 대해 구하여 하나의 표로 정리하여 인식과정에서 활용한다.

학습과정을 통해 구해진 각 문자의 분할과 관련된 통계를 정리하면 표1과 같다. 이 표는, 예를 들어 문자 'm'의 경우 예상했던 것처럼 1개의 세그먼트로 구성된 경우는 전혀 없으며 2-3개의 세그먼트로 구성된 경우는 전체학습문자의 78.7%임을 보여준다. 이 표를 기준으로 하여 각 문자에 대하여, 사용된 학습문자의 98%를 포함하는 세그먼트의 수를 인식과정에서 사용될 가변정합구간으로 정하여 표에서 Δ 로 표시하였다. 즉, 문자 'M/m'의 경우에는 가변정합구간이 4가 되며, 비교적 간단한 형태의 문자인 'I/i' 또는 'L/l'의 경우 가변정합구간은 2가 된다. 여기서 강조되어야 할 사항은, 문자 분할에 대한 통계를 얻기 위하여 단어 영상

을 사용한다는 점이다. 즉, 이미 분할된 문자영상에 분할 알고리즘을 적용하는 것이 아니라, 단어영상에 문자 분할 알고리즘을 적용하여 학습용 문자를 얻음과 동시에 문자분할과 관련된 통계를 구함으로써 얻어진 가변구간의 의미가 실제 인식단계에서 같은 의미를 갖고 적용되도록 한다.

표 1. 각 학습문자의 분할 통계

Table 1. Character segmentation statistics.

	세그먼트의 수				Δ	세그먼트의 수				Δ	
	1	2	3	4		1	2	3	4		
A/a	49.5	95.2	99.7	100.0	3	N/n	3.6	79.7	98.0	100.0	3
B/b	35.8	81.1	98.1	100.0	3	O/o	77.6	98.4	100.0	100.0	2
C/c	83.6	98.8	100.0	100.0	2	P/p	43.2	93.8	99.1	100.0	3
D/d	36.6	90.0	98.9	100.0	3	Q/q	50.0	80.0	100.0	100.0	3
E/e	79.8	97.3	99.5	100.0	3	R/r	66.0	95.9	99.5	100.0	3
F/f	62.1	93.9	99.7	100.0	3	S/s	68.3	95.9	99.9	100.0	3
G/g	39.3	90.0	98.0	100.0	3	T/t	69.4	95.9	98.7	100.0	3
H/h	11.2	79.0	96.2	100.0	4	U/u	1.7	96.9	99.9	100.0	3
I/i	94.3	99.3	100.0	100.0	2	V/v	5.1	94.5	99.6	100.0	3
J/j	51.7	84.5	100.0	100.0	3	W/w	0.0	2.8	88.7	100.0	4
K/k	18.0	78.2	96.7	99.9	4	X/x	18.4	80.8	99.2	100.0	3
L/l	86.9	98.3	100.0	100.0	2	Y/y	7.7	85.0	98.4	100.0	3
M/m	0.0	8.4	79.7	100.0	4	Z/z	54.5	90.9	100.0	100.0	3

3. 초기 정합단계에서 사전활용

문자경계 및 문자정합 결과의 애매모호성이 필기 단어인식과정에 존재하기 때문에 사전의 역할이 중요하다. 문자중심의 단어인식 방법에서는 각 문자세그먼트 또는 문자세그먼트들의 조합의 인식결과를 종합하는 후처리과정에서 사전을 활용하여 사전의 각 단어를 구성하기 위한 비용이 가장 적은 경우를 유사도가 가장 큰 단어로 인식한다. 이를 위해 각 문자세그먼트 또는 그 조합들은 문자인식기를 통해 인식되며 인식의 결과들을 저장-최상의 인식결과 뿐 아니라 후처리 과정에서 의미 있는 단어를 만들기 위해 여러 인식결과를 저장-한다.

반면, 본 논문에서 제안하는 단어중심 단어인식 방법은 인식의 초기단계에서 사전을 활용한다. 앞에서 설명한 것처럼 대부분의 응용분야에서 사전이 제공되며, 특히 사전의 크기가 그리 크지 않은 경우, 사전에 나타난 단어만을 중심으로 정합을 수행하여, 모든 문자 class에 대해 정합을 시도하여 결과를 저장/정리하는 문자중심 단어인식 방법에서 있을 수 있는 불필요한 계산과정을 크게 감소시킬 수 있다.

또한 사전을 인식의 초기단계에 활용함으로써 학습 과정에서 구한 가변구간 개념의 도입도 가능해진다. 즉 분할된 조각의 수, 비교하려는 사전단어의 구성문자수, 그리고 그 구성문자들의 가변구간 등을 통해 실제 사전단어의 각 문자에 대한 정합범위가 동적으로 결정된다.

입력 단어영상과 관련하여 공급되는 사전 내의 단어들을 L_i 라 할 때

$$L_i = \{l_{i0}, l_{i1}, \dots, l_{i(C_i-1)}\} \quad 1 \leq i \leq L_N \quad (4)$$

로 표현하며, 여기서, C_i 는 i 번째 사전단어가 포함하는 문자의 수이며 L_N 은 사전에 포함된 단어의 수이고, l_{ij} 는 i 번째 단어의 j 번째 문자를 나타낸다.

4. 정합과정

필기 단어인식의 문제를 다음과 같이 재정리할 수 있게 된다. 하나의 단어영상이 입력으로 제공되고, 입력영상과 관련된 사전이 공급됨으로 사전에 있는 단어들과 입력 단어영상과의 유사도를 구한다. 이 과정에서, 각 사전 단어를 구성하는 문자들과 초과 분할된 문자 세그먼트들과의 효율적인 정합을 위해 문자 세그먼트의 수, 비교중인 사전 단어를 구성하는 문자의 수, 그리고 학습과정에서 구한 각 문자의 가변 구간 등을 활용하여 단어 정합과 관련된 계산량을 최소화한다.

본 논문에서 제안하는 단어인식 방법의 목적은 입력 단어영상과 사전의 단어 중 가장 잘 정합하는 단어를, 몇 개의 가능한 문자 세그먼트 조합으로부터 추출한 특징벡터와 code words의 기준 특징벡터들을 비교하여 찾는 것이다. Code words는 문자 단위로 학습되었고 하나의 문자영상은 최대 4개의 세그먼트로 구성될 수 있다고 가정하였기 때문에, 하나 또는 몇 개의 세그먼트 조합이 기준이 되는 문자의 code words와 조합이 가능한 구간 내에서 비교된다. 각 정합과정에서 최소거리 값을 내는 경우를 찾아 그 결과를 저장하고, 이 결과를 이용하여 제2단계 정합과정에서 동적 프로그래밍을 사용하여 전체적으로 가장 좋은 정합결과를 내는 단어를 찾는다. 주어진 입력 단어영상 S 에 대해 가장 작은 비교거리(D^*)를 갖는 사전단어 L_i (L^*)를 식(5)를 이용하여 구한다.

$$\begin{aligned} D^* &= \min_{1 \leq i \leq L_N} D(S, L_i) \\ L^* &= \arg \min_i D(S, L_i) \end{aligned} \quad (5)$$

단어 인식을 위한 정합은 두 단계로 나누어 수행된다.

(1) **제1단계 정합:** 비교중인 사전단어 L_i 의 각 구성 문자(l_{ij} , $0 \leq j < C_i$)와 식(3)에서 정의된 단어영상을 구성하는 임의의 세그먼트 구간사이의 정합을 수행한다. 이 때, 비교중인 세그먼트 구간의 시작 세그먼트 번호를 b 로, 마지막 세그먼트 번호를 e 로 표시하고 각 구간의 비교 결과는 식(6)처럼 표현한다.

$$\widehat{D}(v, b, e) = \mathbf{d}(\mathbf{F}(S(b, e)), \mathbf{k}_{i_1}(v)), 0 \leq v < K_{i_1} \quad (6)$$

여기서 $\mathbf{d}(\cdot, \cdot)$ 는 두 특징 벡터사이의 거리이며, $\mathbf{F}(\cdot)$ 는 주어진 세그먼트 구간에서 특징 벡터의 추출을 의미한다. 비교중인 사전 단어의 문자 v 에 대해 가장 잘 정합된 구간을 식(7)을 사용해 구한다.

$$\overline{D}(b, e) = \min_{1 \leq v \leq K_{i_1}} \widehat{D}(v, b, e) \quad (7)$$

정합중인 사전단어에 포함된 하나의 문자와 세그먼트들의 의미 있는 배열과 이에 따른 계산량을 줄이기 위해 문자분할과 관련된 제한사항들, 즉, 사전단어에 포함된 문자의 수, 입력영상의 세그먼트 수, 그리고 각 문자의 가변구간 등을 사용한다. 이 요소들을 종합하여, 비교중인 문자 l_{ij} 에 해당하는 정합창(matching window)의 크기를 정할 수 있게된다. 문자 l_{ij} 에 대해 정합 창의 크기를 결정하는 e 의 범위를 식(8)과 같이 표현할 수 있다.

$$\begin{aligned} e_{\max}^i &= \min(S_N - (C_i - j), (j+1) \cdot \Delta(j) - 1) \\ e_{\min}^i &= \max(S_N - (C_i - (j+1)) \cdot \Delta(j) - 1, j) \\ & \quad j = 1, \dots, L_N - 1 \end{aligned} \quad (8)$$

여기서 $\Delta(j)$ 는 표1에서 제공되는 현재 사전단어의 j 번째 문자의 가변구간이다.

이 구간은 문자분할과 관련된 가정들에 의해 결정된다. 즉, 사전의 단어 L_i 를 구성하는 문자들은 최소한 하나이상의 문자 세그먼트와 정합되며, 하나의 문자는 표1에서 구해진 통계에 따라 대부분의 경우 최대 $\Delta(j)$ 개의 세그먼트로 분할된다. e_{\min}^i 과 e_{\max}^i 사이 구간은 문자 l_{ij} 의 정합창의 크기를 결정하는데 사용된다. 주어진 e 의 범위에 대해서, 다음 문자에 대한 시작 세그먼트 b 는 e_{\min}^i 보다 커야한다. $\widehat{D}(v, b, e)$ 와 식(7)은 이 정합창 내에서만 계산하면 되므로 전체적으로 수행되어야 하는 계산량이 현격하게 감소한다.

각 문자의 가변정합구간은 이와 같은 정합 범위의 제한 이외에, 일부 사전 단어를 일찍 비교 대상에서 보다 엄격하게 제외시키는데 사용될 수 있다. 만일 사전 단어 L_i 에 대해 식(9) 조건 중 하나를 만족하면 그 단어 L_i 는 1단계 정합 이전에 비교대상에서 제외된다.

$$\begin{aligned} S_N &> \sum_j \Delta(l_{ij}) \\ S_N &< L_N \end{aligned} \quad (9)$$

(2) 제2단계 정합: 첫 단계에서 구한 각 정합결과를 종합하기 위해 동적 프로그래밍을 사용하여 전체 단어에 대한 정합결과를 구한다. j 번째 문자에 대해, e 번째 세그먼트에서 끝나는 가장 잘 정합된 경로는 식(10)에 의해 구해진다.

$$\bar{D}_j(e) = \min_{1 \leq b \leq e} [\bar{D}(b, e) + \bar{D}_{j-1}(b-1)] \quad (10)$$

식(10)을 반복적으로 사용하여, 전체적으로 가장 잘 정합되는 경로를 찾는 동적 프로그래밍의 단계들을 다음과 같이 정리한다.

▷ 초기화

$$\bar{D}_0(e) = \bar{D}(0, e) \quad 0 \leq e \leq M_d - 1$$

여기서 M_d 는 하나의 문자를 구성할 수 있는 최대 세그먼트 수이며, 문자 분할 과정에서 'm'과 'w'와 같은 복잡한 문자들을 기준으로 4로 정의하였다. 이 초기화 식에서 볼 수 있는 것처럼, 앞에서 정의한 각 문자의 가변정합구간 개념을 단어의 첫 문자에 대해서는 적용하지 않았는데 이는 일반적으로 필기단어에서 첫 문자는 화려하게 쓰여지는 경향이 있어 대부분의 경우 표1의 통계를 벗어나기 때문이다.

▷ recursion

$$\begin{aligned} \bar{D}_1(e) &= \min_b [\bar{D}(b, e) + \bar{D}_0(b-1)] \\ \bar{D}_2(e) &= \min_b [\bar{D}(b, e) + \bar{D}_1(b-1)] \\ \dots \\ \bar{D}_j(e) &= \min_b [\bar{D}(b, e) + \bar{D}_{j-1}(b-1)] \\ \dots \\ \bar{D}_{C_{j-1}}(e) &= \min_b [\bar{D}(b, e) + \bar{D}_{C_{j-2}}(b-1)] \end{aligned}$$

▷ final minimum distance

$$D^* = \bar{D}_{C_{j-1}}(S_N - 1)$$

이상의 과정을 'tide'라는 단어 영상에 대해 그림3을 이용하여 설명할 수 있다.

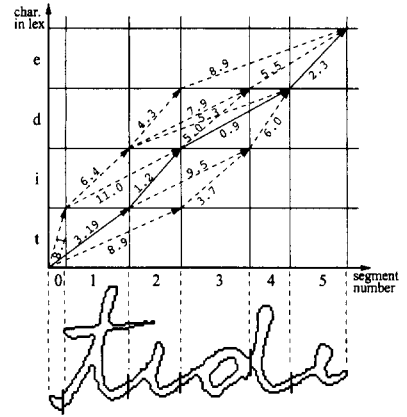


그림 3. 동적 프로그래밍을 이용한 단어의 인식과정
Fig 3. Word recognition procedure using DP.

단어의 분할된 결과를 횡축에 그 번호와 함께 표시하였고, 사전의 단어 중 단어 영상과 일치하는 단어에 대한 1차 정합과정 및 결과를 그림에 표시하였다. 그림에서 화살표는 식(6)에 의한 해당 격자를 구성하는 세그먼트 또는 그 조합과 사전단어의 해당 문자와의 정합 결과를 나타낸다. 예를 들어, 종축의 첫 문자 't'와 횡축의 세그먼트 번호 0에서 2에 해당하는 세그먼트 조합 사이의 정합결과는 8.9이다. 그림의 단어와 주어진 사전단어와의 비교에는 그림에 표시된 화살표 수만큼의 정합이 필요하다. 실선으로 표시된 화살표는 2차 정합과정에서 전체적으로 정합거리를 최소화하는 경로를 찾아 표시한 것이다. 그림의 예에서는 각 문자의 가장 잘 정합된 경우의 연결이 최소 정합경로와 일치하고 있지만, 만일 일부 문자의 정합 결과가 신뢰할 수 없을 만큼 불확실한 경우에는 2차 정합과정에서 주변의 비교적 확실한 정합결과를 갖는 문자들을 중심으로 전체 정합 거리를 최소화하는 경로를 찾게 된다. 이 그림에서 사전단어의 문자 중 'i'와의 비교를 보면, 다른 문자들과는 달리 1차 정합은 1개 또는 2개의 세그먼트 조합에 대해서만 수행되고 있음을 볼 수 있다. 이는 'i'의 가변구간이 2이기 때문이다.

위에서 설명한 정합과정이 사전의 각 단어에 대해 반복됨으로, 사전단어의 특정 위치에서의 어떤 문자에 대한 1단계 정합결과를 저장하여 두면, 다른 사전단어가 그 위치에서 같은 문자를 포함하고 있을 경우 반복되는 계산을 피할 수 있다는 장점을 취할 수 있게 된

다. 예를 들어, 그림1의 경우, 주어진 입력단어 영상에 대해, 사전단어 'east'에 대한 정합과정이 끝난 후, 다음 사전단어 'west'에 대한 정합과정이 수행될 때 두 단어의 공통 부분 'st'에 대한 부분과 'e'에 대한 부분적인 정합과정은 이미 수행되어 생략될 수 있다.

V. 실험 결과

이상에서 설명한 방법을 이용하여 필기단어 인식시스템을 구성하였다. 입력영상은 하나의 독립된 필기단어를 포함하며, 입력영상의 내용을 포함하는 세 가지 크기의 사전을 사용한다. 인식 시스템은 C로 작성되었고 128MByte의 주 메모리와 350MHz의 동작 주파수에서 동작하는 Pentium-II PC에 Linux 2.0을 탑재한 컴퓨터에서 수행하였다. 본 논문에서 도입한 가변구간, 반복계산 감소, 인식결과의 선택여부 결정 등을 중심으로 실험을 수행하여 그 영향을 확인한다.

표 2. 사전 크기에 따른 기능 단위별 평균 처리 소요시간 및 인식률

Table 2. Processing time of each functional unit and recognition rate based on the lexicon size.

구분	기능 단위	10	100	1000	
처리 시간 (msec)	image 변환	2.60		2.78	
	전처리	0.97		1.05	
	문자분할	0.67		0.73	
	특징추출	4.13		4.58	
	정합	가변구간	10.50	39.92	60.15
		고정구간	14.20	49.66	72.42
계	가변구간	18.87	48.29	69.29	
	고정구간	22.57	58.03	81.56	
인식률 (%)	가변구간	1st	96.8	88.6	74.2
		2nd	98.7	93.5	84.1
	고정구간	1st	96.5	87.7	72.5
		2nd	98.7	92.7	82.3

실험에 사용된 영상은 실제 우편물 주소영상에서 분리된 도시, 거리, 사람, 회사 이름들로 구성된 3,000개의 단어 영상이다. 각 단어영상은 크기가 10, 100 및 1,000인 사전들과 함께 제공되는데, 이 사전들은 실제 우편번호를 사용하여 동적으로 얻은 것이다.

1. 가변구간

제4절에 소개된 각 문자의 가변구간이 인식속도 및 정확도에 미치는 영향을 살펴보기 위한 실험을 수행한다.

표2는, 각 사전의 크기에 대해서, 각 기능 모듈의 처리 소요시간과 인식률, 본 논문에서 소개하는 가변구간과 구간의 크기를 4로 고정할 경우에 대해서 각각 보여주고 있다. 가변구간 개념의 적용이 처리 소요시간 및 인식률 향상에 크게 기여하고 있음을 확인할 수 있다.

정합구간을 4로 고정할 경우 가변구간의 정합 범위를 모두포함하기 때문에 가변구간에 의한 정합 방법과 같거나 우수한 결과를 낼 것으로 생각할 수 있으나, 여유 있는 정합구간이 실제 정합과정에서 혼란을 야기할 가능성을 증대시켜 전체적인 인식률은 감소하게 된다. 그림4에는 이처럼 고정정합구간 개념으로는 인식되지 않았던 단어가 가변구간 개념의 적용으로 인식되는 경우의 예를 보여 준다.

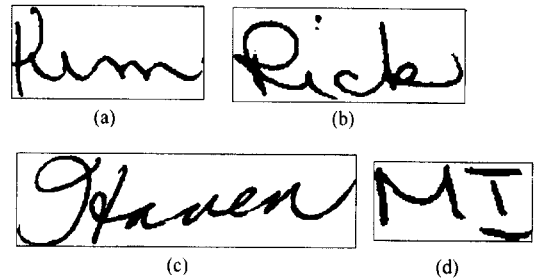


그림 4. 가변 정합구간의 영향을 확인할 수 있는 예 a-b) 고정정합구간이 3일 경우와 c-d) 고정정합구간이 4일 경우

Fig. 4. Sample cases that made difference with the variable duration: a-b) comparing to the fixed duration of 3 and c-d) comparing to the fixed duration of 4.

그림4(a) 및 그림4(b)에서 볼 수 있는 것처럼 'i'와 같이 단순한 모양을 갖는 문자와 'm' 또는 'k'와 같이 복잡한 모양을 갖는 문자들에 대해 정합구간을 모두 3으로 고정하면 단순한 문자는 주변의 세그먼트와의 조합을 통해 사전내 다른 문자에 대해 더 가까운 유사도를 제공할 수 있는 가능성을 증가 시키게되고, 복잡한 모양의 문자들에 대해서는 정합구간을 실제 세그먼트수 보다 작게 제한하여 완전한 문자 정합을 불가능하게 한다. 실제 그림4(a)와 그림4(b)의 두 단어영상에서 'm'과 'k'는 4개의 세그먼트로 분할되어 제공된 사전내의 다른 단어와 더 가까운 유사도를 갖도록 정합

되었다.

그림4(c)에 보인 단어는 문자 정합구간을 4로 고정하였을 때와 가변정합구간을 적용하였을 때, 단어 영상과 일치하는 사전단어 "haven"에 대해 같은 유사도를 얻었으나, 정합구간을 4로 고정하였을 때, 사전내의 다른 단어("chula")에 대해, 근사한 차이지만, 더 근접한 유사도를 보인 경우이다.

그림4(d)의 경우에는 가변정합구간을 적용하였을 때가 고정정합구간 4를 적용하였을 때보다 낮은 유사도를 낸 경우이다. 문자 'i'가 일반적으로 대부분 2개 이하로 분할된다는 통계를 벗어나는 예외적인 경우이다.

2. 반복적인 정합과정 제거

본 논문에서 제안하는 단어 인식 방법의 또 다른 특징은 사전에 포함된 단어들을 인식의 초기 단계에서 활용하여, 문자인식 중심의 단어 인식방법에서 일반적으로 수행되는 후처리 과정을 생략하는 것이다. 이러한 사전 활용방법에 따라, 세그먼트 또는 그 조합을 문자인식 모듈에 보내는 대신 사전단어에 포함된 문자들에 대해서만 제한적인 문자 정합을 수행할 수 있게 된다. 이 과정에서, 어떤 세그먼트 또는 그 조합과 사전 단어의 한 특정 문자간의 정합이 한번 수행되면, 다른 사전단어가 포함하는 같은 문자에 대해서는 정합을 반복할 필요가 없도록, 정합결과를 저장한다. 이처럼 사전중심의 정합을 수행하는 경우 사전의 크기에 따라 크게 증가하는 정합과 관련된 계산량을 감소시킬 수 있다.

표 3. 정합결과와 저장 유무에 따른 정합시간 비교 - 단위는 msec

Table 3. Comparison of time taken for matching with and without caching the intermediate results -in msec.

구분	10		100		1000	
저장 여부	○	×	○	×	○	×
가변정합여부	10.50	13.66	39.92	172.5	60.15	1,049.5
고정정합구간(4)	14.20	20.14	49.66	260.0	72.42	1,540.2

표3은 문자 정합 시, 제안하는 방법을 사용한 경우와 그렇지 않은 경우에 대한 정합소요시간을 가변정합구간과 고정정합구간에 대해 비교하여 나타낸다. 예상할 수 있는 대로 사전단어의 수가 증가 할 경우 정합에 소요되는 시간이 급격하게 증가함을 볼 수 있다.

가변정합구간 개념의 적용과 반복되는 문자 정합결과와 저장으로 사전의 증가에 대한 정합시간의 증가는 표2와 표3에서 볼 수 있는 것처럼 로그함수적으로 증가하는 것을 볼 수 있다. 만일 문자 정합결과를 저장하지 않으면 사전단어 수의 증가에 따라 정합시간도 비례하여 증가함을 표3에서 확인할 수 있다.

3. 신뢰도

표2의 각 사전 크기에 대한 인식률은 인식결과가 입력영상의 내용과 일치하는지 여부를 가려 계산한 결과이다. 실제 응용에서는 단어인식기의 이와 같은 인식결과를 그대로 사용하지 않고 유사도의 임계값들을 적용하여 신뢰할 수 없는 인식결과들을 버리는 방식으로 전체 시스템의 허용오차 범위를 유지하도록 노력한다. 실용적인 측면에서 인식률보다는 신뢰도가 더 중요한 변수가 될 수 있다. 신뢰도는 거부(reject)되지 않고 처리된 단어 중 바르게 인식된 단어의 비율로 표시한다. 본 논문에서는 1차 선택단어의 유사도(Th_1)와, 1차 선택단어의 유사도와 2차 선택단어의 유사도(Th_2)의 차(Th_{D12})를 이용한 인식결과 선택방법을 사용한다. 별도의 단어들에 대한 인식결과를 사용하여 100%의 인식결과를 얻기 위한 변수들을 결정하였을 때 $Th_1 = 5.0$, $Th_{D12} = 0.87$ 로 구해졌다. 이 변수들을 사전의 크기가 10이고 가변정합구간을 사용하여 얻은 인식결과에 적용하였을 때, 2,348개의 경우가 선택되었고 이 중 2,346개 경우가 바르게 인식되어 78.3%의 선택에 대해 99.9%의 신뢰도를 보였다. 참고로 각각의 선택 조건만을 적용하여보면, Th_1 이 5.0보다 작은 조건을 만족하는 경우는 2,658 (88.6%)개의 단어가 선택되어 2,624 (98.7%)개의 단어가 바르게 인식되었고, Th_{D12} 가 0.87보다 큰 조건만을 만족하는 경우는 2,486 (82.9%)개의 단어가 선택되어 2,478 (99.68%)개의 단어가 바르게 인식되었다.

1차 선택단어의 유사도 뿐만 아니라 2차 선택단어의 유사도와의 차이를 함께 고려하여 인식결과에 대한 신뢰도를 효율적으로 검증할 수 있음을 보여준다. 응용분야의 특성, 입력 영상의 특징, 사전의 크기, 기대 신뢰도 등을 고려하여 임계값을 선정한다.

VI. 요약 및 결론

필기 단어의 인식과 관련된 문제는, 기본적으로 여러 원인에 의한 문자 모양의 다양성에 대한 해석과 주

어진 단어를 구성하는 문자들로의 분리 및 정합과정에서의 조합과 관련된 문제들로 나누어 볼 수 있다. 본 논문에서는 영문 필기단어 인식과 관련된 이러한 문제들의 해결을 위한 접근 방법들을 소개한다.

필기단어를 포함하는 대부분의 응용분야들이 입력영상과 관련된 사전이 제공될 수 있고, 단어인식의 문제를 입력영상과 사전의 단어들과의 정합을 통해 가장 유사도가 큰 단어를 선택하는 것으로 해석할 수 있는 경우가 대부분이다. 단어영상을 문자 세그먼트로 분할한 후, 각 세그먼트 또는 그 조합을 문자인식기를 통해 해석하고 그 결과를 제공되는 사전의 단어들과 비교하는 후처리 과정을 거치는 일반적인 접근 방법 대신, 사전을 단어인식을 위한 정합의 초기 단계에 활용하여 인식과정에서 가장 긴 처리시간을 갖는 인식과 관련된 과정을 효율적으로 처리할 수 있게 한다.

제안하는 방법의 장점중의 하나는 그 확장성이다. 본 논문에서는 문자만으로 구성된 단어에 대해서 설명하고 있으나, 만일 문자와 숫자가 혼합된 형태의 단어의 처리도, 관련된 사전만 제공된다면, 제안된 단어인식과 같은 방법으로 처리할 수 있도록 쉽게 확장할 수 있다. 예를 들어 숫자를 포함한 상품코드(예: RJ45, ITB3)와 같은 단어들을 처리하기 위해 기본 알파벳 이외의 각 숫자와 관련된 학습결과(code words 및 가변정합구간)를 인식과정에 추가하여 문자의 경우와 같이 처리한다. 문자인식 중심의 단어처리 방법에서 인식 대상 class의 추가는 인식을 저하와 연결되는 경우가 많지만 제안하는 방법에서는 class를 추가하여도 기존의 문자로만 구성된 단어의 인식률에는 영향을 주지 않고 확장할 수 있다는 장점을 갖는다.

개발된 단어인식 알고리즘은 실제 응용분야에 탑재를 전제로 빠른 처리속도를 내도록 설계되었다. 따라서 문자 분할 및 인식을 위해 정의된 특징들이 추출하기 수월한 것으로 구성되어있다. 이 들 특징의 정제과정을 통해 단순히 문자인식의 연장으로서의 단어인식이 아닌, 단어로부터 직접 구할 수 있는 특징들을 정의하여 인식의 신뢰도를 향상시키는 추가적인 연구가 필요하다.

참 고 문 헌

[1] S. N. Srihari and E. J. Kuebert, "Integration of Hand-Written Address

Interpretation Technology into the United States Postal Service Remote Computer Reader System," in *4th Int. Conf. on Document Analysis and Recognition*, pp. 892-896, 1997.

[2] E. Augustin, O. Baret, D. Price, and S. Knerr, "Legal Amount Recognition on French Bank Checks Using A Neural Network-Hidden Markov Model Hybrid," in *Proc. of Sixth Int. Workshop on Frontiers in Handwriting Recognition*, pp. 45-54, 1998.

[3] 김 기철, 이 성환, "서식 문서 영상의 구조 분석," *한국정보과학회논문지*, vol. 22, no. 1, pp. 182-192, 1995

[4] G. Dzuba, A. Filatov, D. Gershuny, and I. Kil, "Handwritten Word Recognition - The Approach Proved by Practice," in *Proc. of Sixth Int. Workshop on Frontiers in Handwriting Recognition*, pp. 99-111, 1998.

[5] H. Sakoe, "Two-level DP-matching-A Dynamic programming-based pattern matching algorithm for connected word recognition, *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 27, pp. 588-595, Dec. 1979.

[6] K. M. Sayre, "Machine Recognition of Handwritten Words: A Project Report," *Pattern Recognition*, vol. 5, pp. 213-228, 1973.

[7] S. Madhvanath, *The Holistic Paradigm in Handwritten Word Recognition and Its Application to Large and Dynamic Lexicon Scenarios*. PhD thesis, Dept. of Computer Science, State University of New York at Buffalo, 1996.

[8] L. R. Rabiner and B. Juang, *Fundamentals of Speech Recognition*, New Jersey: Prentice Hall Signal Processing Series, 1993.

[9] 신 봉기, 김 진형, "은닉 마르코프 모델네트워크에 의한 온라인 흘림 필기 한글 인식," *한국정보과학회논문지*, vol. 21, no. 9, pp. 1737-1745, 1994

[10] M.-Y. Chen, A. Kundu, and J. Zhou,

“Off-Line Handwritten Word Recognition Using a Hidden Markov Model Type Stochastic Network,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 16, pp. 481-496, May 1994.

[11] H. Hild and A. Waibel, “Connected Letter Recognition with a Multi-State Time Delay Neural Network,” in *Advances in Neural Information Processing System 5*, pp. 712-719, 1993.

저 자 소 개



金 敬 煥(正會員)

1984년 서강대학교 전자공학과 졸업 (공학사). 1986년 서강대학교 전자공학과 졸업(공학석사). 1996년 미국 뉴욕주립대(SUNY at Buffalo) 전기 및 컴퓨터공학과 졸업(공학박사).

1986년 ~ 1995년 금성전기/정밀 기술연구소 선임연구원. 1993년 ~ 1997년 8월 CEDAR/SUNY 연구원. 1997년 9월 ~ 현재 서강대학교 전자공학과 조교수. 주 관심분야는 문서영상처리, 컴퓨터비전, 패턴인식