

---

# 실시간 처리를 위한 컨테이너 ISO코드 인식시스템의 구현

최태완\*

## Implementation of the Container ISO Code Recognition System for Real-Time Processing

Taewan Choi\*

### 요 약

컨테이너 ISO코드 인식시스템은 ISO코드 검출 및 영상 획득, ISO코드 영역 추출, 개별 문자 추출, 문자인식 및 데이터베이스의 5가지 핵심부분으로 구성된다. 이 중에서도 ISO코드 추출의 정확성은 전체 시스템 인식률에 지대한 영향을 줄 수 있는 부분이며, 다양한 컨테이너 종류 및 주위 환경변화에서도 정확한 추출을 요구한다. 본 논문에서는 획득된 영상을 주위 환경변화에도 적응 가능한 이진화 방법을 사용하여 ISO코드 템플릿의 영역을 이진화하고 ISO코드의 분포를 가지는 후보 영역을 추출한다. 추출된 후보 영역 중에서 ISO코드 문자 분포의 특성을 이용한 검증과정을 통해 최종 영역을 추출하여 ISO코드를 인식하는 시스템을 설계 및 구현하였다. 구현된 시스템을 실시간으로 컨테이너에서 획득한 영상에 적용한 결과 다양한 컨테이너 종류 및 주위 환경변화에서도 ISO코드 영역이 정확히 추출됨을 확인하였다.

### ABSTRACT

This paper describes system to extract ISO codes in container image. A container ISO code recognition system for real-time processing is made of 5 core parts which are container ISO code detection and image acquisition, ISO code region extraction, individual character extraction, character recognition and database. Among them, the accuracy of ISO code extraction can affect significantly the accuracy of system recognition rate, and also the more exact extraction of ISO code is required in various weather and environment conditions. The proposed system produces binary of the ISO code's template regions using an adaptive thresholding, extracts candidate regions containing distribution of ISO code, and recognizes ISO codes as detecting a final region through the verifications by using character distribution characteristics of ISO code among the extracted candidates. Experimental results reveal that ISO codes can be efficiently extracted by the proposed method.

### 키워드

Container ISO code, number plate recognition, image processing, projection, segmentation, linear interpolation

## I. 서 론

차량 번호판이나 컨테이너 ISO코드 인식시스템은 ITS (intelligent traffic system)의 한 핵심 분야이며 사람의 시각과 기억력을 보조하는 수단으로 많이 연구되고 있으며

일부 시스템들이 자동화를 위해 설치되어지는 추세에 있다[1]. 영상처리를 이용한 차량 번호판 인식시스템은 번호판을 추출하고 인식하는데 사용되어진다. 또한 이 시스템은 도난 차량의 추적, 통행료 징수 및 주차 관리의 자동화, 방문차량의 자동 확인, 고속도로 톨게이트에서 수

배차량의 인식과 일정 구간에서 차량의 통행 시간과 통행량 조사 등에 광범위하게 응용될 수 있다.

컨테이너 ISO코드 인식시스템은 차량 번호판 인식시스템과 유사성을 가지고 있어 조합해서 사용될 경우에 장점이 많다[2]. 현재 일부 항만에서 컨테이너의 자동화 처리를 위해 채택되고 있는 시스템은 바-코드(bar-code) 시스템, RFID 방식, 영상처리를 기반으로 하는 컨테이너 ISO코드 인식시스템으로 구분된다. 바-코드 시스템은 컨테이너에 바-코드를 부착하거나 소지하여 처리하는 것으로 바-코드의 훼손이나 분실 시에 컨테이너를 자동으로 처리할 수 없을 뿐 아니라 ISO코드(식별자)가 변경될 때마다 바-코드를 교체해야 하는 문제점을 갖고 있다. 따라서 이 시스템은 추가 경비지출이 계속하여 발생하고 관리비용의 과다 지출로 인하여 오늘날 항만에서의 컨테이너 처리를 위한 시스템으로는 영상처리를 이용한 컨테이너 식별자 인식시스템을 적용하고자 하는 추세에 있다.

기존의 연구 방법은 크게 밝기값 기반의 방법과 특징기반의 방법이 있다[3],[4]. 밝기값 기반의 방법은 영상의 밝기값 정보를 이용하여 번호판 영역의 일정 변화를 이용하여 번호판 영역을 추출하는 방법으로 번호판 영역 테두리의 훼손에 영향을 적게 받으나, 주변 날씨와 환경 변화에 따른 잡음에 의해 오인식되는 경우가 많은 것이 단점이다[5]. 반면에 특징기반의 방법은 에지를 구하고 번호판 테두리의 수평 및 수직성분을 허프 변환[6]을 이용하여 차량 번호판 영역을 추출하는 방법으로 처리시간이 길며, 번호판 테두리 훼손에 따른 오인식이 많다[7],[8]. 이외에 번호판이나 ISO코드 인식은 신경회로망을 이용하여 처리하고자 하는 연구도 있었다[9].

컨테이너 식별자의 문자 유형에는 어떤 표준이나 강제적인 규약이 없으므로 어떤 문자는 어떤 폰트(font)를 사용한다고 한 마디로 단언할 수 없다. 따라서 식별자들은 어떤 규정을 제외하고 특별히 정형화되어 있지 않기 때문에 인식하는데 어려움이 있고, 모든 문자들은 컨테이너의 외부에 적혀 있기 때문에 문자의 유실이나 오물 등과 같은 잡음으로 인하여 문자를 인식하는데 어려움이 따른다. 그리고 문자가 표기된 컨테이너 바탕 부분의 굴곡으로 인하여 문자의 왜곡과 같은 변형과 명암 등으로 인한 문자의 잡음 개입과 유실의 영향을 받을 수 있다. 또한 차량에서 획득한 영상은 차종, 컨테이너 형태, 다양한 날씨와 주위 환경변화에 따라 다르다[10],[11].

본 논문에서는 항만에서 컨테이너의 선적 및 야적 등

과 같은 물류처리를 자동화하기 위한 컨테이너 ISO코드에 대하여 매우 가변적이고 다양한 날씨와 주위 환경변화에도 정확히 ISO코드를 인식하는 실시간 처리 시스템을 구현하고자 한다. 구현된 컨테이너 ISO코드 인식시스템은 차량 검출 및 영상 획득, ISO코드 영역 추출, 개별 문자 추출, 문자인식 및 데이터베이스 구축의 5가지 핵심부분으로 구성된다. 이 중에서도 ISO코드 영역 추출의 정확성은 전체 시스템 인식률에 지대한 영향을 주는 핵심부분이다.

## II. 컨테이너 검출 및 영상 획득

컨테이너 ISO코드 인식시스템은 외부 하드웨어 설비와 이를 제어하는 시스템 및 획득 영상으로부터 번호판과 ISO코드를 인식하는 부분으로 구성된다. ISO코드 인식에 필요한 외부 하드웨어 설비는 철근 구조물, 포토센서, 카메라 및 Strobe로 구성된다. 그림 1과 같이 게이트에 진입하는 컨테이너 차량의 인지는 게이트 측면의 6개의 포토센서에 의해 감지된다. 개별 카메라에는 야간이나 어두운 환경에서 밝은 영상을 획득하기 위해 Strobe인 조명장치 부착된다.

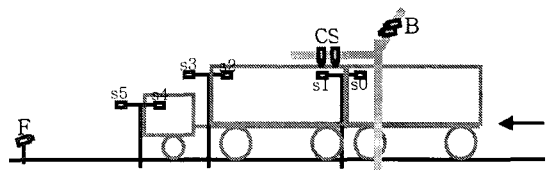


그림 1. 컨테이너 차량 검출 및 영상획득 시스템.  
Fig. 1. Detection of container vehicle and image acquisition system.

시스템 제어부는 영상획득부, 렌즈 autoIris 및 센서 검출부, CdS 및 Strobe부와 자료전송부로 나뉜다. 영상획득은 Matrox사 Metro II/MC grabber 보드를 사용하였다[2]. 렌즈 autoIris 및 센서 검출보드는 PC의 ISA보드에 설치하여 사용한다. 센서는 게이트에 진입한 차량의 위치에 따라 필요한 부분의 영상을 획득하기 위한 트리거(trigger) 신호로 사용된다. ISA autoIris 및 센서 보드는 CCD 카메라의 autoIris 렌즈를 제어하는 부분과 외부입력 센서의 값을 획득하는 부분으로 구성된다. CdS 및 Strobe 부분은 카메라가 설치된 주변 밝기값을 모니터링하여 야간이나 어

두운 상태에서 조명을 작동시키는 역할을 한다. 코드 인식부에는 번호판과 ISO코드를 인식하는 프로그램이 설치된다. 인식 프로그램은 영상획득, 초기설정, 번호판 인식, ISO코드 인식, 번호판 코드 학습 코드, ISO코드 학습 코드를 처리하는 과정이 포함되어 있다.

### III. 컨테이너 ISO코드 인식 알고리즘

컨테이너 ISO코드 인식시스템은 전처리, ISO코드 영역추출, ISO코드 추출, ISO코드 인식 과정으로 나누어진다. ISO코드 영역추출은 시스템에서 가장 중요한 부분이며 인식률을 결정하게 된다. 제안된 ISO코드 인식시스템의 입력은 컨테이너 측면과 후면에서 획득된 영상이고, 최종 출력은 인식된 코드가 된다. 컨테이너 측면은 컨테이너 화물의 길이에 따라 한 번 혹은 두 번의 영상획득을 수행한다. 화물의 길이가 20피트 한 개인 경우는 한번의 영상획득을, 20피트 두 개 혹은 40피트 한 개인 경우는 두 번의 영상획득 과정을 거친다.

#### 3.1 전처리

카메라로부터 획득된 영상은 실제 사용하기에 곤란한 잡음을 포함하고 있으므로 이를 제거하고 필요한 정보만을 획득하기 위해 그림 2와 같이 전처리를 한다.

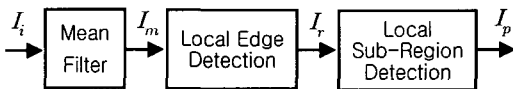


그림 2. 전처리 과정.  
Fig. 2. Preprocessing procedure.

잡음제거는 간단한 선형 필터중 하나이면서 각 픽셀의 값이 지역적으로 이웃한 모든 값의 평균으로 대신하는 지역 평균 연산으로 처리한다. 로컬 부 영역(local sub-region) 검출부에서는 잡음이 제거된 영상을 일정한 크기의 로컬 부 영역으로 나누고, 각 부 영역으로부터 영상의 밝기값 평균(m)과 표준편차(sd), 각 부 영역별 표준편차의 평균( $\sigma$ )을 다음 식에 의해 구한다.

$$m = \frac{1}{N} \sum I(x,y)$$

$$sd = |I(x,y) - m|$$

$$\sigma = \sqrt{\frac{1}{N} \sum (m - I(x,y))^2}$$
(1)

여기서 I와 N은 영상의 밝기값과 크기, x, y는 영상의 위치이다. 그리고 표준편차의 평균값을 임계값으로 하여 밝기값 변화가 적은 부 영역을 다음 식에 의해 제거한다.

$$I_r = \begin{cases} 1 & \text{if } (sd > \sigma) \\ 0 & \text{otherwise} \end{cases}$$
(2)

로컬 에지 검출부에서는 로컬 부 영역을 검출한 후 검출된 영역에서 에지를 구한다. 에지는 ISO코드가 존재하는 부분에 선명하게 나타나며 이를 근거로 위치를 찾게 된다. 에지는 물체의 크기, 형태, 상대적인 위치 등을 분석하고 인식하는데 기본적인 요소가 될 뿐만 아니라 영상을 동일한 영역으로 분할하는 데에도 사용된다. 본 논문의 에지검출에는 다음 식과 같은 Sobel 연산자의 크기를 사용하였다[12].

$$g = \sqrt{G_x^2 + G_y^2}$$
(3)

여기서  $G_x, G_y$ 는 x, y 방향의 연산자를 수행한 결과이다.

#### 3.2 ISO코드 영역추출

전처리된 영상은 ISO코드가 존재하는 영역과 경계가 뚜렷한 영역이 흰색으로 표시되어져 있다. 에지로 판별된 이 영역들 중에서 실제 코드에 대한 영역을 추출해야 한다. ISO코드가 존재하는 영역은 직관적으로 에지들이 일정한 특징을 가지고 모여져 있는 현상을 알 수 있다. 따라서 한 라인씩 스캔하면서 ISO코드 간격이 유지하고 있는 부분의 에지를 추출해낸다. 그림 3은 ISO코드 영역추출 과정을 나타낸다.

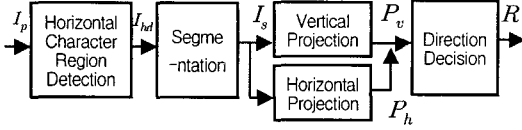


그림 3. ISO코드 영역 추출.  
Fig. 3. ISO code region detection.

3.2.1 수평코드 영역추출

수평코드 영역추출은 한 수평 라인씩 에지와 에지의 간격이 ISO코드 간격으로 일정한 경우와 군집 형태로 뭉쳐져 있는 경우를 발견하고 영역을 표시하게 된다. 선택되는 에지가 ISO코드인 경우는 “//\” 형태를 가진다. 따라서 한 라인에서 “//\” 형태를 가지는 영역을 먼저 찾게 되고, 발견된 영역들이 군집 형태로 뭉쳐져 있다면 이들 ISO코드 후보영역으로 간주한다. 수평코드 영역추출에서는 언급한 “//\”의 형태를 취하는 영역을 찾는 과정에서 수평선과 같은 형태는 제거된다.

3.2.2 영역분할과 프로젝션(projection)

수평코드 영역을 분리한 후에는 영상 전체에서 어디에 ISO코드가 존재하는지를 확인해야 한다. 코드 영역분리는 region-growing 방법을 사용하며 영상의 전 영역 R을 n 개의 하위영역  $R_1, R_2, \dots, R_n$  으로 분할한다[12]. 이에 따라 영역을 분할한 후 남아있는 최종 영역은 ISO코드로 간주되며, 그 위치를 결정하기 위해 수직과 수평 프로젝션을 한다. 에지에 해당하는 값을 x 방향으로 값을 누적하는 수직 프로젝션과 y 방향으로 값을 누적하는 수평 프로젝션은 다음 식과 같이 수행한다.

$$p_v(y) = \sum_x I_s(x, y) \tag{4-1}$$

$$p_h(x) = \sum_y I_s(x, y) \tag{4-2}$$

3.2.3 방향 검출(direction decision)

수직과 수평 프로젝션을 한 후에는 어떤 방향으로 ISO코드가 존재하는지를 판별한다. 수직 프로젝션으로 ISO코드 영역의 분포를 확인하고 수직 프로젝션이 아닌 경우에 수평 프로젝션을 확인한다. 둘 다 아니면 ISO코드가 존재하지 않는 것으로 판별한다. ISO코드는 위치가 다양하고 정해진 기준이 없기 때문에 여러 가지 조건을 고려하

여 다음과 같이 판별한다.

1. 수직 프로젝션을 수행한 후 판별 조건은
  - 임계값을 만족하는 ISO코드의 높이는 25~65 사이로 한다.
  - 만족하는 높이를 가지는 영역이 3개를 넘지 않는다.
  - ISO코드 영역이 하나 존재할 경우는 일정 이상의 너비를 가져야 하고 위, 아래에 존재하는 폭이 65 이상인 영역은 둘로 나눈다.
  - ISO코드 영역이 두 개 이상 존재할 경우 두 개의 영역 조건은 만족하고 세 번째 영역이 일정 거리 이상 떨어져 있으면 ISO코드 영역은 두 개인 것으로 간주한다.
2. 수평 프로젝션 수행 후 판별 조건은 일정 길이 이상의 높이를 가지고 폭이 측정된 ISO코드 범위이내이면서 최고인 지점을 선택한다.

3.3. ISO코드 추출

컨테이너 식별자는 Owner Code(OC: 4개의 영문자), Serial Number(SN: 6개의 숫자)와 Check Digit(CD: 1개의 숫자)로 구성되어 있다. 그러나 식별자의 크기, 두께, 폰트, 색깔 등이 일정하게 규정되어 있지 않고, 또한 외부 환경에 직접 노출되어 있어서 문자의 유실과 변형, 날씨의 변화에 따른 잡음 등이 심하며, 컨테이너 표면의 굴곡에 의한 명암차와 왜곡이 발생한다. 이는 식별자를 인식하는데 많은 문제점을 야기한다. 이런 조건에서 ISO코드의 추출과정은 반전 모드와 비반전 모드, 식별자 위치에 따른 개별문자의 분류, 식별자 모드선택, 반전 모드와 비반전 모드의 결정, 개별문자 추출의 단계를 거친다. 본 논문에서는 컨테이너 식별자를 영상에서 효율적으로 분리하기 위해 다양한 컨테이너 식별자의 형태를 분석한 결과 식별자의 위치에 따른 분류를 함으로써 개별문자를 추출하였다. 컨테이너 영상은 그림 4와 같이 어두운 배경에 밝은 문자(비반전 모드)의 식별자와 밝은 배경에 어두운 문자(반전 모드)의 식별자로 구분된다. 반전 모드와 비반전 모드는 영상의 이진화에서 인식하고자 하는 문자와 배경을 구분하는데 중요한 판단 요소이다.

3.3.1 개별문자 추출

현재 ISO코드는 OC, SN, CD가 표현된 영상에서 위치에 따라 8가지 형태로 나올 수 있고, 이 형태에 따라 영역 내 추출될 문자의 개수를 정보로써 저장할 수 있으며 추

출한 문자가 오류없이 추출되었는가의 판단 요소로 사용할 수 있다. 개별문자 추출부에서는 문자영역 추출부에서 찾은 영상의 수평 또는 수직 영역을 기반으로 컨테이너 식별자를 개별 문자로 분리한다. 먼저 입력으로 주어지는 수평(또는 수직) 영역에서의 에지 영상을 프로젝션 기법을 이용하여 영상에서 식별자가 있는 영역만을 분리한다. 분리된 식별자의 영역을 미리 정해놓은 8가지 모드에서 해당되는 모드를 선택하고, 이 모드의 식별자가 반전인지 비반전인지를 결정한 후 개별문자를 추출하게 된다. 개별문자의 추출은 프로젝션과 레이블링 기법을 사용한다. 추출된 개별문자가 정해진 모드에 일치하는지를 확인한 후 개별문자를 보정한다.

일반적으로 문자 분리는 원 영상을 이진화한 영상에서 프로젝션 또는 레이블링에 의해 이루어진다. 그러나 컨테이너 영상은 식별자의 반전 모드와 비반전 모드가 존재하기 때문에 반전, 비반전 정보가 주어지지 않으면 이진화가 불가능하다. 본 논문에서는 반전과 비반전 모드에 상관없이 식별자 위치를 찾는 방법이 필요하여 영상의 에지 정보를 이용하였다. 그림 5는 프로젝션에 의한 식별자 영역 분리의 세부과정을 나타낸다.

컨테이너 후면 영상에서 수직으로 된 긴 철제 봉이 에지 영상에서 문자와 구분이 잘 되지 않아 문자로 오인될 가능성이 많다. 이 점을 해결하기 위해 에지 영상에서 수직 방향의 긴 라인 에지를 사전에 제거한다. 적정 문자영역의 분리는 전처리된 에지 영상에 대해서 수직 프로젝션을 다음 식과 같이 수행한다.

$$proj[c] = \sum_{r=0}^{height-1} edge[r][c] \tag{5}$$

여기서  $c=0,1,\dots,width-1$ ,  $edge[r][c]$ 는 에지 영상의  $r$ 행  $c$ 열의 값을 나타내며 에지이면 1, 아니면 0이다. 프로젝션에서 적정 문자 후보영역(CR)은 다음의 단계에 의해 찾는다.

**Step 1.** 프로젝션에서 0이 아닌 위치가 검출되었다면, 그 다음 0인 위치를 찾는다.

**Step 2.** 프로젝션의 폭이 끝날 때까지 Step 1을 반복 수행한다.

프로젝션에서 잡음제거는 후보 영역중에서 영역들 간의 간격이 좁은 것을 문자군으로 합쳐서 처리한다. 그리고 적정 문자 후보 영역중에서 잡음제거에 의해 제거되지

않고 남은 영역은 문자가 존재하는 영역으로 간주하여 최종 문자 적정영역으로 분리한다.

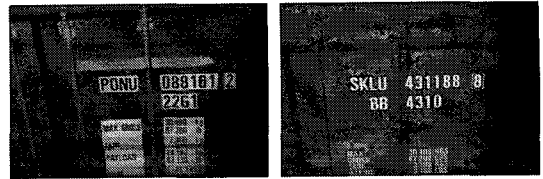


그림 4. (a)와 (b) 식별자의 반전 모드와 비반전 모드의 영상.

Fig. 4. (a) and (b) Inversed and non-inversed mode's images of identifier.

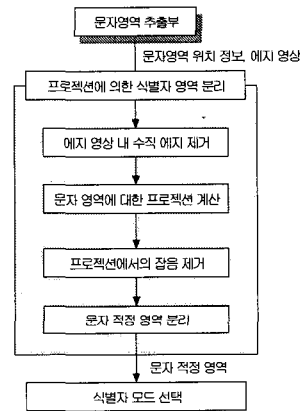


그림 5. 프로젝션에 의한 식별자 영역분리.

Fig. 5. Region segmentation of Identifier by projection.

### 3.3.2 식별자 모드 선택

분리된 문자영역은 개별문자 추출에서 해당 영역내에 몇 개의 문자가 존재하는지를 알기 위해 식별자 위치에 따른 형태적 분류의 8가지 모드 중 해당 모드를 선정하고, 식별자가 몇 행에 걸쳐 존재하는지를 판단해야 한다. 일반적으로 식별자가 한 행에 존재하는 경우, 두 행에 걸쳐 존재하는 경우, 세 행에 걸쳐 존재하는 경우가 대부분이며 세 행을 초과하지 않는다. 판단 기준은 적정 문자영역의 폭과 높이의 비(ratio)로 판단하며 계산식은 다음과 같다.

$$ratio1[j][i] = \frac{TR[j][i].right - TR[j][i].left}{TR[j][i].bottom - TR[j][i].top} \tag{6}$$

$$ratio2[j] = \sum_1^{count[j]} ratio1[j][i]$$

여기서  $j = 1, \dots, M$ ,  $i = 1, \dots, count[j]$ 이다.  $j$ 는 행 번호,  $i$ 는 해당 행에 대한 적정 문자영역의 인덱스를 나타낸다. 또한  $count[j]$ 는  $j$ 행에서의 적정 문자영역의 개수를 나타낸다. 적정 문자영역의 영상에서 반전, 비반전 모드는 그림 6의 과정에 의해 결정된다.

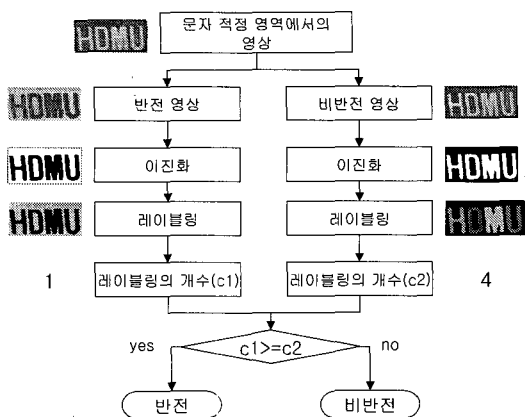


그림 6. 반전, 비반전 모드 결정.

Fig. 6. Decision of Inversed and non-inversed mode.

### 3.3.3 개별 문자추출

반전, 비반전 모드의 정보와 최종 문자영역, 추출해야 할 문자 개수를 기초로 개별 문자를 추출한다. 개별 문자 추출은 반전과 비반전 모드의 영상 특징이 다르기 때문에 각 모드별로 과정을 다르게 한다. 본 논문에서 비반전 모드의 개별 문자추출은 프로젝션 기법을 기준으로 레이블링을 추가적으로 사용하였고, 반전모드는 레이블링을 기준으로 프로젝션 기법을 추가하여 사용하였다. 일반적으로 비반전 모드의 식별자는 문자간의 간격이 존재하여 프로젝션 기법으로 개별 문자를 추출할 수 있다. 전처리된 이진영상에서 자간 간격이 존재하므로 개별 문자 추출이 가능하다. 그러나 반전 모드는 비반전 모드와 달리 문자영역 외의 외곽 영역에도 이진화 영상의 문자와 같은 값을 가지게 된다. 이로 인해 프로젝션에 의해 자간 간격 차에 의한 분리는 불가능하기 때문에 레이블링에 의해 개별 문자를 추출한다.

### 3.4 ISO코드 인식

ISO코드의 최종 인식은 전처리로 코드를 먼저 정규화와 선형화한 후 ISO코드의 대표패턴을 생성해야 한다. ISO코드를 추출한 각 개별 문자는 위치와 크기가 일정하

지 않으므로 인식하는데 어려움이 있어서 각 개별 문자들을 일정한 위치와 크기로 고정시킨 후 Mesh를 통해 특징값을 추출하여 인식시킨다.

#### 3.4.1 ISO코드의 정규화

ISO코드를 추출한 각 개별 문자는 ISO 추출 영역범위에 따라 크기와 위치가 달라지므로 일정한 크기와 위치로 고정하기 위해 정규화한다.

ISO코드는 총 11개의 문자로 앞에 네 자는 대문자 A~Z 사이이고, 뒤의 일곱 자는 0~9사이의 숫자로 이루어져 있다. 이것의 ISO코드별 크기는 그림 8과 같다. 각 개별 문자는 바탕은 흰색, 문자는 검정색이다. 바탕이 아닌 문자만을 추출하여 ISO코드별 크기에 따라 고정한다. 즉, 검정색이 있으면 문자로 보고 세로방향으로 검정색이 처음 시작되는 위치와 끝나는 위치를 찾아 문자만 추출한 영상으로 크기를 고정시킨다. 가로방향은 ISO코드를 추출한 가로길이를 그대로 사용한다. 크기를 일정하게 고정시키는 계산식은 다음과 같다.

$$Factor_{width} = 1.0 - (w_c/w_t) \tag{7}$$

$$Factor_{height} = 1.0 - (h_c/h_t)$$

여기서  $Factor_{width}$ ,  $Factor_{height}$ 는 크기고정 가로와 세로 인수값,  $w_c$ 는 ISO코드에서 추출한 가로길이,  $w_t$ ,  $h_t$ 는 크기를 고정할 세로와 가로길이,  $h_c$ 는 문자의 끝 위치-시작위치이다. ISO코드가 문자인 경우의 크기고정 인수값은 그림 9와 같다. ISO코드가 숫자인 경우의 크기고정 인수값은 문자와 같으나 단지 가로 크기고정 인수값을 구하는 식에서  $w_t$ 의 값이 "24"가 된다. 그리고 식 (7)을 사용하여 선형 내삽법(linear interpolation)으로 크기를 고정시킨다.

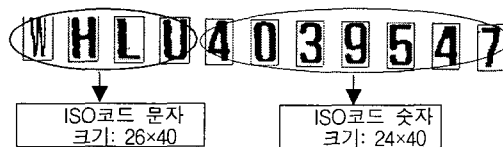
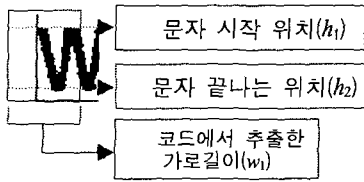


그림 8. ISO코드 문자와 숫자 크기.

Fig. 8. Size of ISO code's characters and numbers.



크기 고정 인수값(가로) =  $1.0 - (w_1 / 26)$

크기 고정 인수값(세로) =  $1.0 - ((h_2 - h_1) / 40)$

그림 9. 크기고정 인수값의 계산.  
Fig. 9. Calculation of size-fixed parameters.

3.4.2 ISO코드 센터화(centering)

크기를 일정하게 고정시킨 각 개별 문자인식을 쉽게 하기 위해서는 문자들을 일정한 크기에 맞게 가운데로 옮겨야 한다. 먼저 세로 방향으로 문자의 시작과 끝나는 위치를 찾아서 상하로 빈 공간이 가운데로 옮길 수 있을 만큼의 공간이 있는지 확인하고 빈 공간이 충분하다면 세로 방향의 가운데로 옮기고, 그렇지 않을 경우는 그 문자의 크기가 크기를 고정하는 길이만큼 있다고 보고 가운데로 옮기지 않는다. 마찬가지로 가로 방향으로 가운데로 옮기는 것은 가로 방향으로 문자의 시작과 끝나는 위치를 찾아서 좌우로 빈 공간이 가운데로 옮길 수 있을 만큼의 공간이 있는지 확인하고 빈 공간이 충분하다면 가로 방향의 가운데로 옮기고, 그렇지 않을 경우는 가운데로 옮기지 않는다.

3.4.3 인식 텍스트 파일작성

이 과정에서는 정규화한 각 개별 문자 영상을 이진 텍스트화된 각 개별 문자를 코드별로 어느 특정 폴더에 저장한다. ISO코드의 대표패턴을 만들 때에는 직접 패턴을 만드는 것보다 ISO코드 영상에서 각 개별 문자를 추출한 영상을 사용하는 것이 인식에 좋으므로 인식하고자 하는 텍스트화된 파일을 대표패턴으로 한다. 특정한 폴더에 대표패턴을 만들고자 하는 텍스트화 되어 있는 문자 파일과 그 파일이 대표패턴으로 사용한다는 것을 알리기 위해 각 코드별로 어느 특정한 파일에 저장한다.

3.4.4 ISO코드의 최종인식

ISO코드의 정규화를 거친 각각의 개별 문자를 Mesh를 통해 특징값을 추출하여 대표패턴과 비교한다. 물론 각 코드별 대표패턴들도 Mesh를 통해 특징값을 추출하여 메

모리에 저장해 둔다. 대표패턴과 비교했을 때 가장 근접한 거리값을 인식하게 된다. ISO코드의 인식과정에서 제일 중요한 것은 Mesh feature인데 간단히 설명하면 다음과 같다. 4x4 크기의 마스크를 사용하여 이진 텍스트의 패턴값을 더한다. 이진 텍스트를 4x4 크기의 마스크를 사용하여 패턴값을 더하면 '1'인 값만 더해진다. 따라서 바탕이 아닌 문자인 것을 '1'로 텍스트화 시킨 것이기 때문에 그 패턴의 특징만을 추출한다. 각 코드별로 대표패턴들을 Mesh feature를 사용하여 특징값을 추출한 후, 메모리에 Mesh 패턴을 저장하고 다른 메모리에는 대표패턴에 의한 목표값을 저장한다.

정규화한 번호판 코드 개별 문자는 '0'과 '1'로 구성된 이진 텍스트이다. 인식시킬 각 패턴들을 Mesh feature를 사용하여 특징값을 추출하고, 추출한 각 개별 문자는 각 코드별로 저장된 대표패턴들의 특징값과 비교하여 최소 거리값을 구하고, 최소거리값을 가지고 있는 대표패턴인 텍스트의 목표값이 인식되어진다. Mesh feature 거리값은 다음 식에 의해 구해진다.

$$\sum_{i=0}^{총패턴수-1} d(i) = \sum_{i=0}^{총패턴수-1} \sum_{j=0}^{1mesh패턴수-1} \sqrt{(I_r(j) - I_p(i,j))} \quad (8)$$

여기서,  $d[i]$ 는 대표패턴 수만큼 개별 문자와 대표패턴 사이의 거리값,  $I_r(j)$ 는 인식시킬 패턴,  $I_p(i,j)$ 는 대표패턴이다. 식 (8)에 의해 구해진 거리값 중에서 최소거리값은

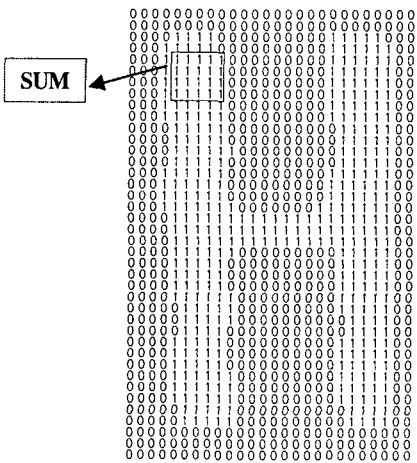
$$Distance_{Min} = Min(\sum_{i=0}^{mesh패턴수-1} d[i]) \quad (9)$$

이다. 여기서  $Distance_{Min}$ 은 최소거리값,  $d[i]$ 는 대표패턴 수만큼 개별 문자와 대표패턴사이의 거리값이다. 그림 10은 이와 같이 처리한 ISO코드 'H'의 인식과정이다. 그림 (c)에서 ISO코드 문자인 'H'는 최소거리값 '1.258'인 대표패턴 인덱스 '2'에 해당되는 목표값 'H'를 인식하게 된다. 각 코드별로 인식을 하는데 대표패턴과 비교하였을 때 최소거리값이 30이상일 경우는 "?"로 표시한다.

IV. 실험 및 결과

제안한 시스템은 디지털 카메라를 사용하여 게이트에 진입하는 컨테이너 차량 앞면의 번호판 영상과 컨테이너 측면과 후면의 ISO코드 영상을 실시간으로 1024×1024 크기로 획득한다. 실시간의 영상획득은 항만에 출입하는 실제 컨테이너 차량을 대상으로 주간과 야간에 이루어졌다. 컨테이너와 차량번호판 인식을 위한 기본적인 데이터의 학습이 종료된 후, ISO코드의 인식은 Pentium 4(3.0GHz) 프로세서에서 Visual C++ 6.0으로 수행하였다. 본 논문에서 전처리 과정의 로컬 부 영역 검출은 잡음이 제거된 영상을 30×30 픽셀(pixel) 크기로 로컬 부 영역의 영상으로 나눈다. 이것의 처리 결과는 그림 11이고, (c)는 식 (3)에 의해 구한 크기들을 평균한 값을 임계값으로 정하여 처리한 결과이다.

수평코드 영역추출은 수평의 한 라인씩 에지와 에지의 간격이 ISO코드 간격으로 일정한 경우와 군집 형태로 뭉쳐져 있는 경우를 찾아 영역을 표시하게 되며, 그림 12(a)에 “//\” 형태를 막대 그래프로 나타내었다. 수평코드 영역추출에서 언급한 “//\” 형태를 취하는 영역을 찾는 과정에서 수평선과 같은 형태는 제거하여 전체 영상에 대해 수평코드 영역추출을 수행한 결과를 (b)에 나타내었다.



(a)

인덱스	0	1	2	3	...	44	45	46	47
Mesh 특징값	0	4	0	0	...	0	0	0	0

(b)

메모리인덱스 대표패턴 인덱스	0	1	2	3	...	44	45	46	47	거리값	목표값
	0	0	2	1	...	1	0	0	0	5.125	
	1	0	3	4	0	...	0	0	0	9.256	F
	2	0	4	0	0	...	5	0	0	1.258	H
⋮	⋮										⋮
176	0	2	2	3		2	0	0	0	27.158	Z

(c)

그림 10. ISO코드 문자 'H'의 인식 과정. (a) ISO코드 문자 'H'. (b) ISO코드 문자인 'H'를 Mesh feature한 메모리 구조. (c) 대표패턴을 Mesh feature한 메모리구조와 ISO코드 문자 'H'를 비교했을 때의 거리값.

Fig. 10. Recognition process of ISO code's character 'H'. (a) ISO code's character 'H'. (b) Mesh-featured memory structure of ISO code's character 'H'. (c) Distance values as comparing with (b) for representative pattern.

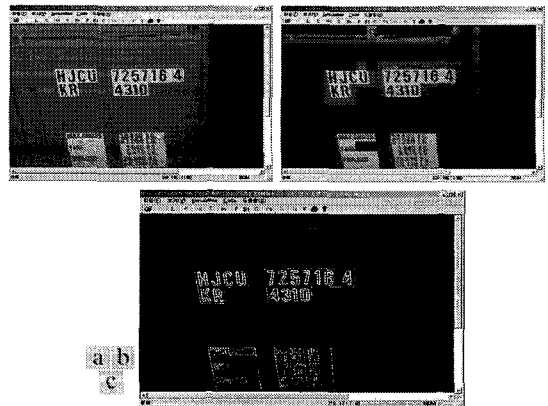


그림 11. 로컬 부 영역 검출 결과. (a)입력영상. (b) 로컬 부 영역 검출 영상. (c) 로컬 에지 검출 영상  
Fig. 11. Results of Local sub-region detection. (a) Input image. (b) and (c) Images by local sub-region and local edge detection.



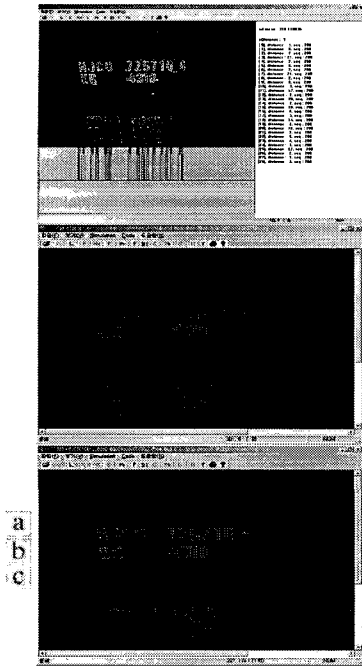


그림 12. ISO코드 영역과 분할결과. (a) 스캔 라인의 수평 코드 영역. (b) 수평 코드 영역 검출. (c) ISO코드 영역 분할.

Fig. 12. ISO code region and segmented results. (a) and (b) Horizontal code region of scan line and its detection. (c) segmentation result.

수평코드 영역을 분리한 영상에서 ISO코드가 어디에 있는지를 알기 위해 region-growing 방법을 사용하여 최종 ISO코드 영역을 분할한 결과는 그림 12(c)이고, 분할 후 남아있는 최종 영역을 코드로 간주하여 수직과 수평 프로젝션을 수행하여 얻어진 영상은 그림 13이다.

프로젝션을 한 다음에는 에지 영상에서 실제 코드의 인식에 장애가 되는 수직 에지를 제거해야 한다. 그림 14(a)는 문자 영역추출부에서 출력으로 주어지는 문자의 수평영역을 나타낸다. 그림 (a)의 컨테이너에 수직의 긴 철재 봉이 (b)의 에지 영상에서 문자와 구분되지 않아 문자로 오인될 수 있다. 이 점을 극복하기 위해서 에지 영상에서 수직 방향으로 긴 라인 에지를 제거시켜 처리한 결과는 그림 (c)이다.

에지 영상에서 수직 라인 에지를 사전에 제거한 후에 적정 문자영역을 분리하기 위해서는 식 (5)를 이용하여 프로젝션을 수행한다. 프로젝션의 결과와 적정 문자 후보 영역을 검출한 결과는 그림 15이다. 그림 14의 에지 영상

내 수직에지 제거에서 완벽하게 긴 수평 라인 에지를 제거시키지 못하고 문자 이외의 영역에서 잡음이 발생함을 볼 수 있다. 즉 그림 15(a)의 1번 영역이다. 프로젝션에서 잡음 제거는 먼저 후보 영역중에서 영역들의 사이 간격이 좁은 것은 문자군으로 합친다. 그림 15(a)에서 후보 영역 3과 4, 4와 5는 그 사이 간격이 좁으므로 하나의 문자군으로 취급하여 얻어진 결과는 (b)이다. 이때 사이 간격에 대한 문턱치는 5 픽셀이다. 문자 후보영역에 대해서 (1) 문자 후보 영역의 폭(CR[i].length)이 5 픽셀 미만인 영역, (2) 문자 후보 영역 중 프로젝션의 최고 높이가 높이의 50% 미만인 영역, (3) 문자 후보 영역에 해당하는 에지 영상에서 그림 16과 같이 프로파일에서 두 개 이상의 돌출이 없는 영역은 잡음으로 간주하여 제거시킨다.

적정 문자 후보영역 중에서 잡음이 제거되지 않고 남은 영역은 문자가 존재하는 영역으로 간주하여 최종 문자 적정영역으로 확정한다. 그림 14(a)의 원 영상에 대한 문자 적정영역 분리 결과를 그림 17에 나타내었다. 분리된 영역은 좌측 상단의 좌표와 우측 하단의 좌표를 버퍼 (TR)에 저장한다. 저장 버퍼 TR[j][i]는 j번째 행에서 i번째 문자 영역을 나타내며 TR[j][i].top은 좌측 상단 y좌표를, TR[j][i].left는 좌측 상단 x좌표를, TR[j][i].bottom은 우측 하단 y좌표를, TR[j][i].right는 우측 하단 x좌표를 저장한다.

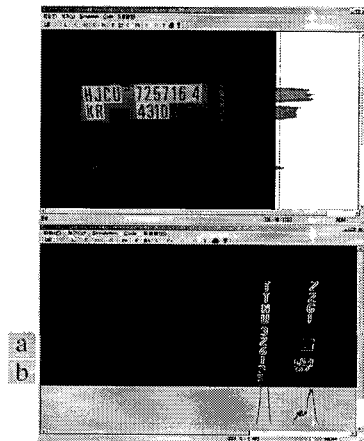


그림 13. (a), (b) 수직과 수평 프로젝션 결과.  
Fig. 13. (a) and (b) Vertical and horizontal projection results.

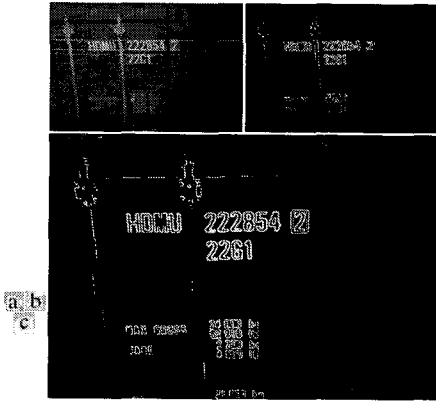


그림 14. (a) 문자의 수평 영역. (b) 에지 영상. (c) 수직 방향으로 긴 라인 에지를 제거시킨 후 영상.  
 Fig. 14. (a) and (b) Horizontal region of characters and its edge image. (c) Results of removing long line edges to the horizontal direction.

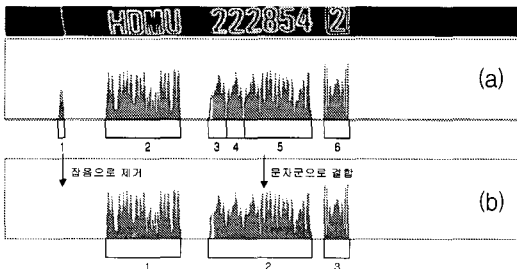


그림 15. 첫 번째 행의 문자영역에 대한 수직 프로젝션과 잡음제거.  
 Fig. 15. Vertical projection for the first row character region and noise reduction.

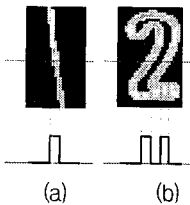


그림 16. (a), (b) 문자와 잡음의 에지 특성.  
 Fig. 16. (a) and (b) Edge characteristics of character and noise.



그림 17. 문자 적정 영역 분리 결과.  
 Fig. 17. Segmented results of the proper character region.

앞서 언급했듯이 비반전 모드의 식별자는 전처리된 이진영상에서 자간 간격이 존재하므로 개별문자의 추출이 가능하다. 그러나 프로젝션했을 때 자간 간격이 없어 분리할 수 없는 경우는 x축으로 문자를 점차적으로 기울임으로써 자간 간격이 최대가 되는 시점에서 프로젝션에 의해 개별문자를 추출할 수 있다. 만약 "5"와 "4"자가 붙어서 x축으로 기울려도 찾을 수 없는 경우는 추출된 개별 문자중에서 x, y 비가 큰 영역을 프로젝션에서 최소가 되는 위치를 찾아 잘라낸다. 반면에, 반전 모드에서 자간 간격차에 의한 분리는 레이블링에 의해 개별 문자를 추출한다. 문자영역에 대한 이진화 처리시 문자의 외곽부에서도 문자와 같은 값을 가지는데 외곽부의 레이블링은 일반적인 문자와 차이점이 있다. 즉, 문자의 레이블링은 폭과 높이의 비율이 0.8미만인데 비해서 외곽부의 레이블링은 0.8이상인 경우가 대부분이므로 레이블링 후 폭과 높이의 비율이 큰 경우는 레이블링에서 제외시킨다. 실험한 결과 레이블링의 수와 추출되어야 할 문자 개수가 일치하는 정상적인 경우는 개별문자가 정확히 추출됨을 볼 수 있다. 그러나 외곽부와 문자영역에 잡음이 섞여서 추출하고자 하는 개수보다 레이블링된 개수가 작은 경우(under segmentation)는 이진화 영상의 프로젝션과 레이블링 영상의 프로젝션의 차 프로젝션에 의해 레이블링되지 않은 곳을 찾아내어 분리시킨다.

개별 문자를 분리하여 ISO코드의 각 개별 문자인식을 쉽게 하기 위해서는 ISO코드를 정규화하고 문자들을 일정한 크기에 맞게 가운데로 옮긴다. 이를 처리한 결과는 그림 18과 같다.

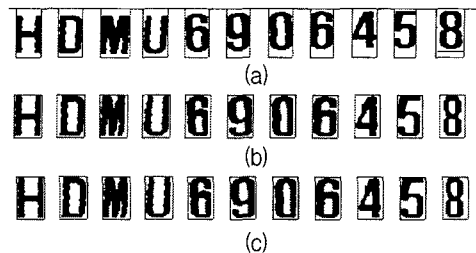


그림 18. ISO 코드 정규화와 센터화 처리 결과. (a), (b) ISO 코드 정규화 전과 후. (c) ISO코드 센터화.  
 Fig. 18. Results of ISO code normalization and centering. (a) and (b) Before and after normalizing of ISO code. (c) its centering.

ISO코드의 센터화가 끝나면 문자인식을 위한 인식 텍스트 파일을 작성하는데 파일의 첫 줄에는 그 문자의 총

패턴 수와 가로길이가 기록된다. 바탕의 흰색은 '0'으로, 검정색은 '1'로 하여 텍스트화시킨 각 개별 문자를 코드별로 어느 특정 폴더에 저장한다. 이 파일의 형태는 그림 19와 같으며, 대표패턴이 저장된 특정 폴더에 각 코드별로 특정한 파일이름에 대표패턴을 추가하는 방법으로써 ISO코드가 문자인 경우 'iso\_char.dat' 파일에 대표패턴이 추가되어진 형태는 그림 20과 같다.

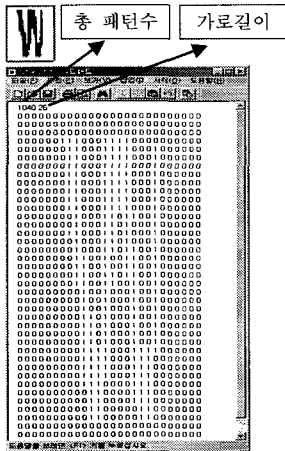


그림 19. ISO코드별 정규화된 영상의 텍스트 파일.  
Fig. 19. A text file of normalized image to ISO code.

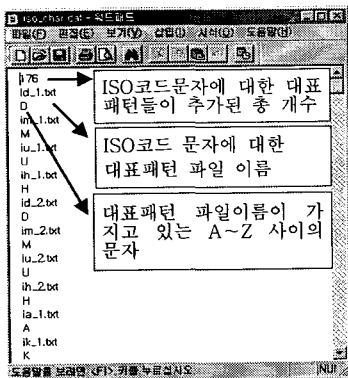


그림 20. 인식 파일작성.  
Fig. 20. A made recognition file.

## V. 결 론

제한한 시스템은 항만에서 컨테이너의 선적 및 야적과 같은 물류처리를 자동화하기 위한 컨테이너 ISO코드에 대하여 매우 가변적이고 주위 환경변화에도 ISO코드를 정확하게 인식하는 실시간 처리용 하드웨어와 프로그램을 탑재하였으며, 컨테이너의 차량 번호판을 인식할 수 있도록 구현하였다. 이 시스템은 센서를 통해 컨테이너 차량을 감지한 후 CCD 카메라를 작동시켜 컨테이너 영상을 고성능으로 획득할 수 있으며, 획득된 영상은 전처리 단계를 거친 후 문자 영역추출, 개별 문자추출, 문자인식을 하는 과정으로 구성되어 있다. 특히, 구현된 시스템은 문자가 표기된 컨테이너의 바탕 부분의 굴곡으로 인하여 문자의 왜곡과 같은 변형과 명암 등으로 인한 문자의 잡음 개입과 유실의 영향을 받을 수 있는 조건에서도 문자를 인식하기 전에 수행되는 프로젝션에 의한 식별자 영역의 분리과정에서 잡음을 제거하기 때문에 우수한 성능을 보이며 인식률을 높이는 장점을 가진다. 그리고 외부 하드웨어 설비의 수정 및 보관을 통하여 자동차 번호판 인식시스템으로도 효율적으로 변환시킬 수 있도록 유연성이 있는 시스템으로 설계하였다.

향후 과제는 어려운 조건의 컨테이너 영상으로부터 ISO코드의 인식을 개선시킬 수 있는 방법과 인식 불가능한 식별자를 위한 인공지능적인 방법을 이용한 문자 인식의 연구가 필요하다.

## ACKNOWLEDGMENTS

Portions of this work were supported by the InPEG Co., Ltd. The author would like to thank CEO, Dr. H.I. Kang and Dr. S.W. Seol for providing data.

## 참고문헌

- [1] Ming G. He, Alan L. Harvey and Thurai Vinay, "Vehicle Number Plate Location for Character Recognition," *ACCV'95 2nd Asian Conf. on Computer Vision, Singapore*, pp. 1425-1428, Dec. 1995.
- [2] 최태완, "실시간 차종인식 시스템의 설계 및 구현," *한국해양정보통신학회논문지*, 10권 5호, 2006.
- [3] W. Hwang and H. Ko, "Real-time Vehicle Recognition Using Local Feature Extraction," *Electronic Letters*, vol. 37, no. 7, pp. 424-425, Mar. 2001.
- [4] Christoph Bush, Ralf Dörner, Christian Freytag, Heike Ziegler, "Feature Based Recognition of Traffic Video Streams for Online Route Tracing," *Proc. of the IEEE Conf. on Vehicle Technology Conf.*, pp. 1790-1794, 1999.
- [5] E. R. Lee, P. K. Kim, and H. J. Kim, "Automatic recognition of a car license plate using color image processing," *IEEE Int'l Conf. on Image Proc.*, vol. 2, pp. 301-305, 1994.
- [6] C. Kimme, D. Ballard and J. Sklansky, "Finding circles by an array of accumulators," *Comm. ACM*, vol. 18, pp. 120-122, 1975.
- [7] Mei Yu and Y. D. Kim, "An approach to Korean license plate recognition based on vertical edge matching," *SMC, IEEE Int'l Conf.* vol. 4, pp. 2975-2980, 2000.
- [8] T. Naito, T. Tsukada, K. Yamada, K. Kozuka and S. Yamamoto, "Robust recognition methods for inclined license plates under various illumination conditions outdoors," *Intelligent Trans. Systems Proc., IEEE/IEEJ/JSAI Int'l Conf.*, pp. 697-702, 1999.
- [9] 이만형, 황대훈, "Back-propagation 알고리즘을 이용한 컨테이너 식별자 인식에 관한 연구," *대한전자공학회 추계종합학술대회 논문집*, 제21권 제1호, Jun. 1998.
- [10] J. M. White and G. D. Rohrer, "Image thresholding for optical character recognition and other applications requiring character image extraction," *IBM J. Develop.*, 1983.
- [11] Yuichi Togashi and Hideaki Yamaguchi, "Stable Position Measurement of a Moving Air plane by Adaptive Camera Control," *MVA'94 IAPR Workshop on Mach. Vision App.*, pp. 13-15, Dec. 1994.
- [12] Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, 2nd Ed., Prentice Hall, 2002.

## 저자소개

## 최 태 완(Taewan Choi)



1996년 부산대학교 전자공학과  
(공학박사)

1985년 부산대학교 전자공학과  
(공학석사)

1983년 동아대학교 전자공학과 (공학사)

1984년 12월 ~ 1991년 2월 (주)LG전자 디지털어플라이언스 연구소 선임연구원(팀장)

1991년 3월 ~ 1993년 2월 부산대학교 전자공학과 조교

1993년 3월 ~ 1997년 2월 부산대학교 전자공학과 시간강사

1996년 3월 ~ 1997년 2월 (주)하나정보기술 부설연구소 기술이사

1997년 3월 ~ 현재 진주산업대학교 메카트로닉스공학과 부교수

※관심분야: 컴퓨터비전 및 영상처리, 신경회로망, 패턴 인식 등.