



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

2017년 2월

석사학위 논문

# 딥러닝에 기반한 다양한 환경에서의 색상 및 문자 인식

조선대학교 대학원

컴퓨터공학과

박 현 철

# 딥러닝에 기반한 다양한 환경에서의 색상 및 문자 인식

Color and Character Recognition Based on Deep Learning  
in Real Environment

2017년 2월 24일

조선대학교 대학원

컴퓨터공학과

박 현 철

# 딥러닝에 기반한 다양한 환경에서의 색상 및 문자 인식

지도교수 이 상 웅

이 논문을 공학석사학위 신청 논문으로 제출함

2016년 10월

조선대학교 대학원

컴퓨터공학과

박 현 철

## 박현철의 석사학위논문을 인준함

위원장	조선대학교	교수	<u>양 희 덕</u> (인)
위 원	조선대학교	교수	<u>문 인 규</u> (인)
위 원	조선대학교	교수	<u>이 상 웅</u> (인)

2016년 11월

조선대학교 대학원

## 목 차

ABSTRACT .....	v
I. 서 론 .....	1
1. 연구 배경 및 기존 연구 .....	1
II. 배경 이론 .....	5
1. 신경망 .....	5
2. 컨볼루션 뉴럴 네트워크 .....	7
1) 컨볼루션 층 .....	8
2) 풀링 층 .....	9
3) 출력 층 .....	10
III. 문자 및 색상 인식 .....	11
1. 문자 인식 .....	11
1) 문자인식을 위한 영상 이진화 방법 .....	11
2) 문자인식 추출 방법 .....	12
3) 제안하는 문자위치 추정 방법 .....	14
4) 문자 인식 .....	16
2. 색상 인식 .....	17
1) 색 분류 .....	17
2) 제안하는 광원 추정을 위한 CNN 구조 .....	19

IV. 실험	20
1. 실험 데이터	20
2. 실험 방법	23
3. 실험 결과	25
V. 결론	31
[참고문헌]	32

## ABSTRACT

# Color and Character Recognition Based on Deep Learning in Real Environment

Hyun-Cheol, Park

Advisor : Prof. Sang-Woong, Lee Ph. D.

Department of Computer Engineering,

Graduate School of Chosun University

Since the previous decade, Deep Learning revolutionized the field of Artificial Intelligence by providing it with an immense performance boost. The performance continues to get better with the increase in amount of data. Deep Learning has been made possible due to the advances in hardware technologies that has enabled fast computation of large number of complex hidden layers in the neural network. The exemplary performance of Deep Learning has been the motivation for its implementation in this thesis.

In this thesis, Deep Learning is applied for Character Recognition in tire images. Previously, the localization of individual characters was done using horizontal and vertical histogram projections of the image. In an attempt to increase the accuracy of localization, this thesis proposes to implement Deep Learning algorithms, particularly Convolutional Neural Network (CNN).



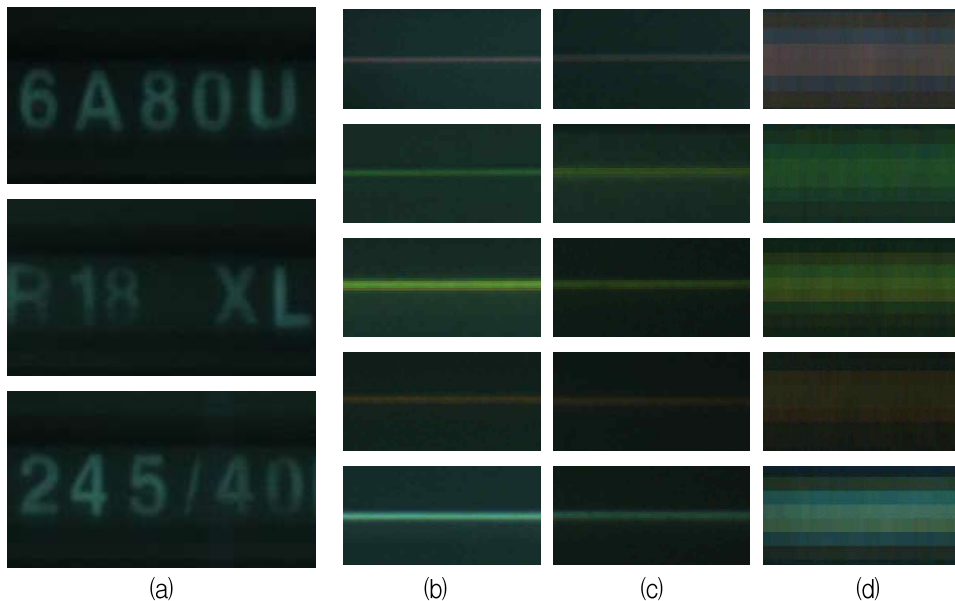
This thesis also classifies color by averaging all pixels in the R, G, and B channel of a color image. Then these three calculated average values are fed into a feed forward neural network for classification. However, this is not enough due to the fact that the color of an object changes with varying illumination conditions. The human color perception has the ability to perceive a relatively constant color despite changing illumination conditions. This phenomenon of identifying the color even with varying illumination is known as Color Constancy. Unfortunately, machines do not recognize this concept of color constancy which leads to an inaccurate color classification. For example, a banana is yellow when the illumination is white but its color changes with the change in the color of illumination. This thesis automatically estimates the color of illumination using CNN.

# I. 서론

## 1. 연구 배경 및 기존 연구

인간의 시각체계는 주변 빛의 세기와 눈의 피로도에 따라 색의 인지능력이 변할 수 있다. 일부 산업체 현장에서는 제품에 출력된 문자열과 특정 색상을 구별하는 반복적인 작업을 요한다. 이러한 검사 작업은 작업자의 눈의 피로도를 쉽게 유발하며 피로 누적에 의해 검사 작업 중 오류가 발생 할 수 있으며 품질 하락을 초래하는 경우가 많다. 이러한 상황을 개선하기 위하여 자동시각인식 검사시스템 활용이 필수적이다.

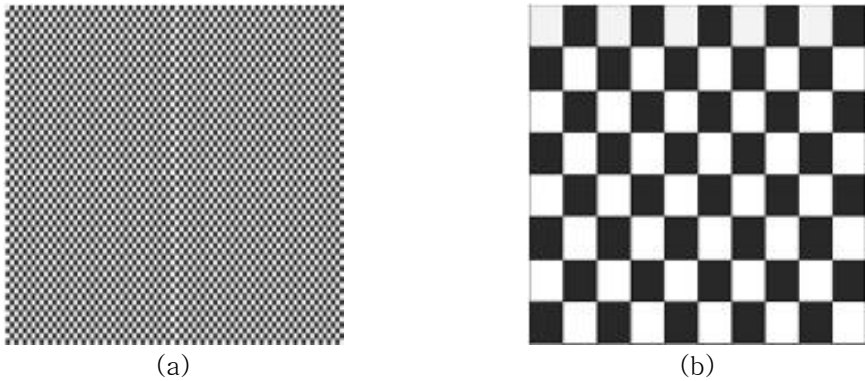
본 논문에서는 제품의 품질 확인을 위하여 제품에 그려진 문자 및 도료의 색상을 구분하는 자동화 문자인식 및 색상인식 시스템을 만들고자 한다. 실험에 사용되는 문자 및 색상은 그림 1과 같이 산업체에서 사용되고 있는 압인 문자와 다섯 가지의 색상을 분류한다.



[그림 1] 압인 문자 및 색선(상단부터 밑으로 빨강, 초록, 노랑, 갈색, 청록)  
 (a) 압인 문자 (b) 정상 색선, (c) 채도와 명도 값이 낮은 색선, (d) 확대 이미지

그림 1의 (b)는 색상의 농도와 빛의 밝기가 적당한 정상 이미지이며 (c)는 도료의

농도가 부족하거나 빛의 밝기가 약한 이미지이다. 색 공간중 하나인 HSV 모델은 Hue (색조), Saturation(채도), Value(명도)의 세 가지 성분으로 색을 표현되는데 채도와 명도의 값이 낮아질수록 색상 차가 근사해져 사람은 정확한 색조 구분이 힘들어진다. 즉, 산업체 현장에서 도료의 농도와 빛의 밝기가 약할 경우 비슷한 색조를 가진 색으로 잘못 인식 할 수 있다. 그 예로 그림 1의 (c) (빨강, 갈색), (초록, 노랑) 이 두 쌍의 색상 구분은 작업자로 하여금 잘못된 판단을 내릴 수 있다. 선행된 개발에서는 명도와 채도를 나타내는 2차원 좌표 상에 다섯 가지의 대표 색상을 나타내고 그림 1의 (d)와 같이 확대된 이미지로부터 표본 픽셀을 추출하여 2차원 좌표에 나타난 대표 색상들 중 유클리디안 거리가 가장 근접한 색상으로 인식하도록 하였다.



[그림 2] 착시 예  
(a) 회색 영상, (b) 회색영상의 확대 이미지

하지만 대표 표본 픽셀을 이용한 색상 추정에는 한 가지 문제가 있다. 그림 2와 같이 흰색과 검정색 픽셀로 이루어진 이미지에서 표본을 추출할 경우, 사람은 (a) 이미지와 같이 회색으로 인지하지만 컴퓨터 시스템에서 색상인식을 위해 대표 표본 픽셀을 추출 시에는 흰색 또는 검정색 픽셀을 추출하여 사람이 인식하는 것과 다른 결과가 나올 수 있다. 이처럼 본 연구에서도 그림 1의 (c)에서와 같이 확대된 영상에서는 순수한 색조를 가진 픽셀뿐만 아니라 먼지, 오물 또는 그늘짐으로 인한 노이즈가 발생하여 다른 색감이 추출될 수 있다. 따라서 착시가 발생하는 이미지에서 명도가 낮은 픽셀을 표본으로 추출할 경우 잘못된 인식 결과를 범할 수 있다.

또한 작업장의 광원이 흰색이 아닌 다른 색상으로 변한 경우인 색 항상성 문제가 남아있다. 광원의 변화로 영상속의 문자 및 색선들이 영향을 받아 색의 변형이 일어나

고유의 색을 인식하기 어려운 문제가 발생한다. 인간은 기본적으로 사물의 색을 인지할 때 광원으로부터 물체가 반사한 반사광을 인지하여 그 사물의 색을 알 수 있다. 만약 광원의 색이 흰색이 아닌 다른 색을 나타낼 때 물체는 고유의 색을 반사하지 못하고 광원의 영향을 받아 다소 변형된 색을 반사한다. 인간은 이 광원의 효과를 배제하는 능력이 뛰어나 고유의 색을 알아낼 수 있지만 영상처리를 통한 광원 배제 처리방법은 다소 어려운 문제에 속한다.

이전의 색 항상성 연구로는 통계적 기반의 방법이 주를 이루었다. 대표적으로 Gray World 알고리즘이 있다. 일반적인 영상에는 다양한 색상이 존재하기 때문에 영상 내의 전체 화소에 대한 평균값이 회색 축에 있을 확률이 높다. 그래서 영상의 모든 픽셀들의 평균이 회색 축에 있다고 가정하고 이미지 내의 모든 화소들의 평균값을 이용하여 이미지의 조명을 추정하고 화소를 보정하는 방법이다[1].

Perfect Reflector 알고리즘은 영상 내의 최대 밝기를 가지는 화소에 영상에 비추지는 광원의 정보가 가장 많이 내포되어 있다는 가정을 한 방법이다. perfect reflector 알고리즘의 광원 추정 방법은 광택 또는 반사되는 표면에 광원의 주요 성분들이 내포되어 있기 때문에 이러한 화소들을 이용하여 영상의 광원을 추정하는 방법이다. 하지만 반사면이 없거나 색 채널이 포화상태일 경우 색상 보정이 불가능한 단점이 있다[2].

Color by Correlation 알고리즘은 사전에 계산된 상관관계 행렬을 만들어 입력 이미지의 광원을 추정하는 방법이다. 일반 광원에서 나타나는 색상과 달라진 광원에서 나타나는 색상 사이의 유사도가 가장 높은 학습 영상의 광원을 입력 이미지의 광원으로 추정한다. 이 방법은 입력 이미지와 학습된 이미지의 색 분포도가 유사할 경우 비교적 정확하게 광원을 추정할 수 있다. 단, 학습 이미지와 입력 이미지의 유사도 및 학습 이미지의 다양성에 상당히 의존적이다[3].

학습기반의 알고리즘에는 Gamut 알고리즘[4], SVR 기반 알고리즘[5], Exemplar 기반 알고리즘[6] 그리고 베이지안 접근법[7] 이 있다. 일반적으로 위와 같은 학습 기반의 알고리즘들은 영상 속의 화소 값과 모서리와 같은 낮은 수준의 영상 특징들을 직접 모델링해야한다. 하지만 이러한 영상 특징들은 본질적으로 계층적이고 이미지 데이터로부터 자동적으로 학습되어 어느 특정 특징 구조로만 편향됨을 피할 수 있다. 그리고 학습기반 알고리즘에 신경망 알고리즘이 있다. 초기의 신경망 알고리즘은 입력층의 뉴런이 매우 큰 크기의 이진화 된 색도 히스토그램으로 구성되었으며 출력층은 두 개의 색도 값으로 광원을 나타내었다. 이러한 방법은 학습시 매우 많은 학습 데이터가 필요

로 하였고, 학습이 정확히 잘 이루어져도 특정한 표면을 나타내는 영상에서만 좋은 결과를 얻을 수 있었다[8][9].

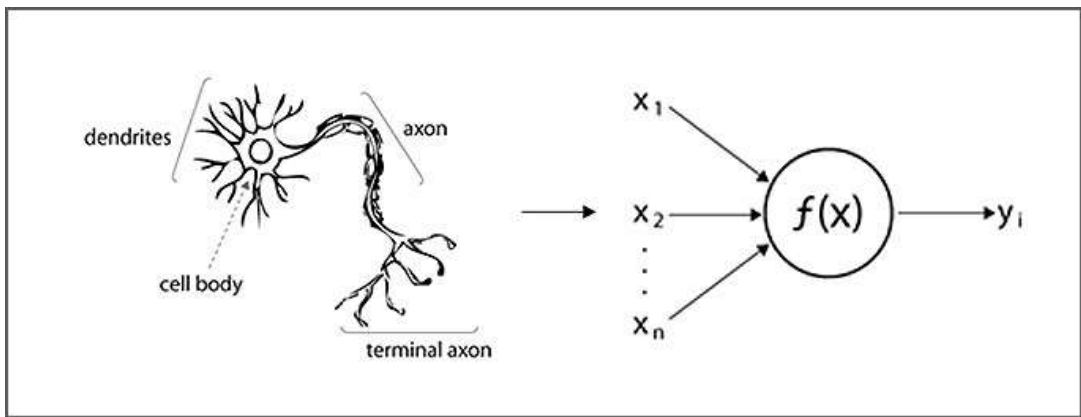
근래에는 기계학습 발달과 빠른 컴퓨팅 기술이 가능해짐에 따라 다양한 딥러닝 방법들이 생겼다[10][11]. 딥러닝 방법 중 하나인 컨볼루션 신경망(Convolutional Neural Network, CNN)은 다양한 문자인식[12] 및 이미지 클래스 분류[13]에서 월등한 성능을 보이고 있다. 이에 Lou[14]는 다섯 개의 컨볼루션층이 포함된 총 8개의 은닉층 구성된 구조를 제안하였다. Lou의 방법은 사전에 낮은 수준의 특징들을 정의하였던 기존에 학습기반의 알고리즘 방법들과 달리 CNN 구조를 사용함으로써 다양한 특징들을 커널 학습을 통하여 광원 추정을 하였다.

본 논문에서도 마찬가지로 CNN을 이용하여 영상의 광원을 추정하고 다양한 조명 환경과 유사한 색상분포를 가지는 색상환경에서 딥러닝 학습을 통한 색상 분류 문제를 해결하고자 한다. 또한 CNN을 통하여 문자영역과 비 문자영역을 추정하는 방법을 제안한다.

## II. 배경 이론

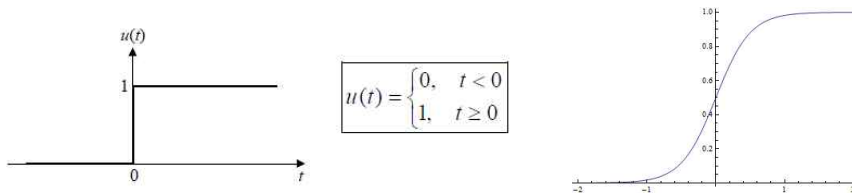
### 1. 신경망

신경망은 1943년 맥클로크(McCulloch)와 피츠(Pitts)가 신경세포를 모델링하여 뉴런이라 불리는 유닛간의 상호 연결 및 네트워크를 구성하여 AND, OR, NOT과 같은 논리 연산이 가능함을 보여주었다. 모델링에는 신경세포의 수상돌기를 입력 유닛으로, 세포체의 가중치는 각 유닛과 연결된 연결 가중치를 곱한 합으로 계산하여 활성화 함수를 통해 출력 값을 얻을 수 있도록 모델링 하였다[15].



[그림 3] 인체의 뉴런과 퍼셉트론

그리고 활성화 함수로는 계단함수, 시그모이드 함수, 쌍곡선 함수, ReLU등이 있다. 계단함수는 입력 값이 문턱치 값 이상이면 출력이 1로 활성화되는 함수이다. 시그모이드 함수는 입력 값의 크기에 따라 0과 1의 한계 값에 가까워지고 쌍곡선 함수는 -1과 1에 가까워지는 활성화 함수이다.

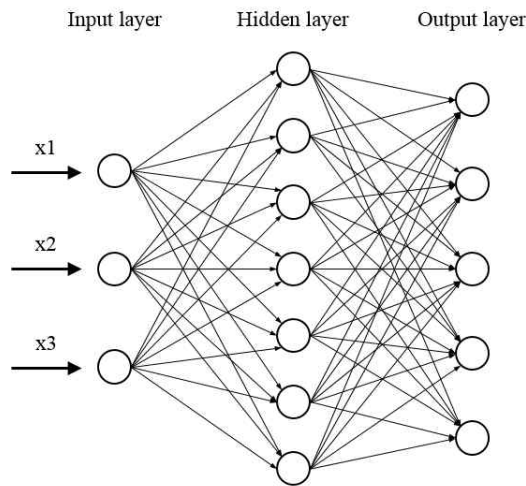


[그림 4] 활성화 함수

이후 Rosenblatt의 퍼셉트론이 발표되며 패턴인식 관련 종사자들에게 큰 관심을 보였다. 퍼셉트론은 훈련 집합에 대해 잘못된 분류결과를 보일 경우 가중치 값을 변화시키며 올바르게 분류될 때까지 반복한다.

$$y(x) = wx + b \quad (1)$$

퍼셉트론의 발표 후 가중치의 학습을 위한 적절한 학습방법의 부재와 Minsky와 Papert에 의해 퍼셉트론은 식(1)와 같이 초평면으로 특징 공간을 두 개의 영역으로 분할하는 것에 불과함을 보이며 XOR문제 해결의 결함이 있음을 증명하였다. 이후 신경망의 침체기가 시작되었으나 오류 역전파 알고리즘과 다층 퍼셉트론의 개발로 선형분리가 불가능한 문제도 해결할 수 있게 되었다. 다층 퍼셉트론은 하나의 층만 가지고 있는 퍼셉트론이 여러 개의 층을 가지는 다층 퍼셉트론으로 진보된 신경망으로 입력층, 은닉층, 출력층을 가진다.



[그림 5] 인공신경망

은닉층의 유닛은 입력층의 유닛들의 가중된 합을 계산하고 활성화 함수를 통하여 출력층으로 전달될 값이 계산된다.

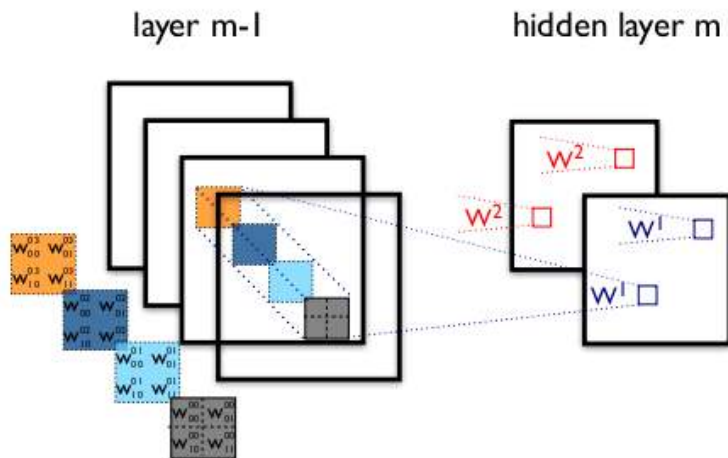
$$hidden\ unit = \sum wx + b \quad (2)$$

최종적으로 출력층의 유닛에는 로지스틱 회귀로 소프트맥스 함수를 가장 많이 사용한다. 식(3)과 같은 소프트맥스 수식에 따라 모든 출력 값들의 합이 1이 되는 일반화의 효과를 얻게 되며 이는 특정 클래스가 나올 확률을 나타내게 된다.

$$P(Y=i|x, W,b) = softmax(Wx+b) = \frac{e^{W_i x + b_i}}{\sum_j e^{W_j x + b_j}} \quad (3)$$

## 2. 컨볼루션 뉴럴 네트워크

컨볼루션 뉴럴 네트워크는 크게 컨볼루션층, 풀링층, 출력층으로 구성된다. 컨볼루션층에서는 영상의 특징을 추출하고 컨볼루션 특징으로 변환하여 풀링층에서 일정 영역 대푯값을 계산하여 사이즈가 감소된 영상을 얻는다. 이는 다시 동일한 구조의 컨볼루션층과 풀링층에 의해 계산되며 여러 층을 거치며 영상의 특징을 추출하고 최종적으로 출력층에 입력되는 구조이다. 그림 6은 컨볼루션 뉴럴 네트워크의 구조를 보여준다[16].

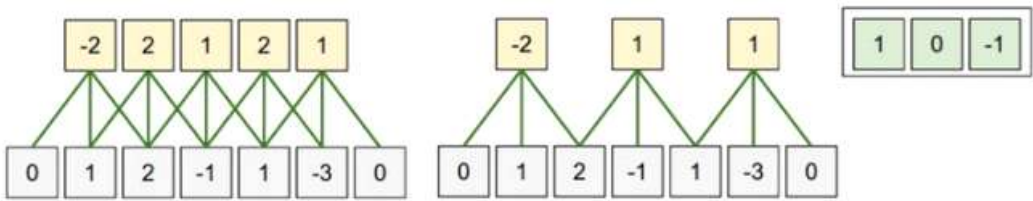


[그림 6] 컨볼루션과 신경망



## 1) 컨볼루션 층

2차원 이미지에 컨볼루션을 수행하는 것은 영상의 특징을 추출하는 과정으로써 2차원 이미지에 일정 크기의 커널을 슬라이드하며 입력 이미지의 픽셀들과 곱하고 이를 모두 더하여 특징을 추출한다. 여기서 커널을 역전파를 통해 학습함으로써 많은 영상 데이터로부터 식별성이 좋은 특징을 뽑아낼 수 있으며, 다양한 커널을 사용하여 이미지의 다양한 특징을 추출할 수 있다. 아래 그림 7은 연산방법의 예이다.



[그림 7] 컨볼루션 예

이때, 컨볼루션 연산 방법에는 보통 3가지가 존재한다. 만약  $N \times N$  크기의 이미지와  $m \times m$  크기의 컨볼루션 커널이 있을 때, 첫 번째의 연산 방법은 커널과 영상이 겹치는 픽셀이 1개 이상일 때 컨볼루션 연산을 수행 하게 된다. 이때 출력 영상의 사이즈는 확대되며 식(4)를 통해 계산 된다.

$$z_{ij} = \sum_{u=0}^{m-1} \sum_{v=0}^{m-1} w_{uv} x_{(i+u+m-1)(j+v+m-1)} \quad (4)$$

두 번째 연산 방법은 커널의 중심이 영상의 픽셀이 존재하는 곳만 일치 시켜 컨볼루션 연산을 수행 한다. 이때 출력 영상의 사이즈는 입력 영상의 사이즈와 같고 아래 식(5)를 통해 계산된다.

$$z_{ij} = \sum_{u=0}^{m-1} \sum_{v=0}^{m-1} w_{uv} x_{(i+u+((m-1)/2))(j+v-((m-1)/2))} \quad (5)$$

마지막으로 커널이 전체가 영상의 픽셀이 존재하는 부분과 겹칠 경우에만 컨볼루션 연산을 할 경우이다. 이때 출력 영상의 사이즈는 감소되며 아래 식(6)을 통해 계산된다.

$$z_{ij} = \sum_{u=0}^{m-1} \sum_{v=0}^{m-1} w_{uv} x_{(i+u)(j+v)} \quad (6)$$

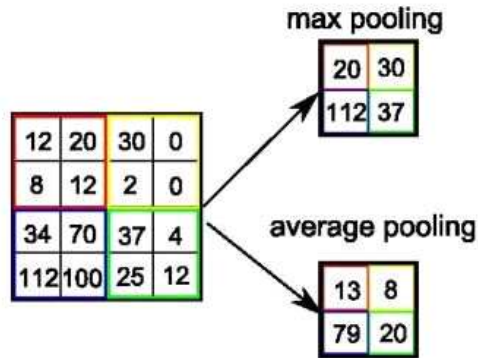
본 논문의 컨볼루션 뉴럴 네트워크는 위의 식(6)을 통해 계산된다. 이와 같은 연산 방법으로 계산된 특징 이미지는 바이어스 값이 더해지고 활성화 함수를 통해 컨볼루션 층에 최종 출력된다. 식은 아래와 같다.

$$h = f(z_{ij} + b) \quad (7)$$

여기서, b는 바이어스를 나타내며 f는 활성화 함수로 사용한 시그모이드 함수를 나타낸다.

## 2) 풀링 층

풀링층에서는 영상의 사이즈를 자연수 배로 축소시키는 과정으로 크게 Max pooling과 Average pooling 방법이 있다. Max pooling은 축소배수와 동일한 사이즈의 커널에 대응되는 픽셀의 값 중 최댓값을 갖는 값을 사용하여 영상의 사이즈를 감소시킨다. Average pooling은 커널에 대응되는 픽셀의 평균값을 계산하여 영상의 사이즈를 감소시킨다. 이러한 부분 추출 과정은 일정 영역을 그 영역의 대표적인 값을 갖는 1x1 사이즈로 축소함으로써 영상의 사이즈를 감소시켜 연산량을 감소시키고 이동, 회전, 크기변화에 불변성을 갖도록 하여 출력의 영향을 줄이는 역할을 한다. 아래 그림 8은 부분 추출의 예를 보여준다.



[그림 8] 최댓값 풀링과 평균 풀링

### 3) 출력 층

출력층은 분류단계에 해당하며 Fully connected layer가 존재한다. 풀링층을 통해 나온 이미지들은 1차원 벡터로 나열되고 output layer와 fully connection으로 연결되어 있으며 output layer는 데이터의 class 수만큼 노드가 존재한다. output layer의 노드들은 아래 식(8)을 통해 계산된다.

$$y_i = f(x_j \times w_{ij} + b_i) \quad (8)$$

여기서, x는 pooling layer를 통해 나온 특징 이미지들을 1차원 벡터로 나열한 것이고 이미지 크기가 N x N이고 n개의 특징 이미지가 있다면  $j=1,2,\dots,N \times N \times n$ 이다. w는 분류의 입력층과 출력층 사이에 연결되어 있는 가중치 값이며  $i=1,2,\dots,m$  (클래스 수) 이다. b는 바이어스이며 f는 활성화함수로 사용된 시그모이드 함수이다. 이렇게 나온 출력 값은 Max값을 찾아 Label을 확인하여 인식을 수행한다.

### Ⅲ. 문자 및 색상 인식

#### 1. 문자인식

본 장에서는 문자 추출방법 및 딥러닝을 사용한 문자인식을 설명한다. 일반적인 문자인식 순서도는 그림9와 같다. 본 장의 구성은 문자인식에 필요한 영상 이진화 방법 및 문자영역 추출에 대하여 설명하고 컨볼루션 뉴럴 네트워크를 통한 문자 영역 추정 방법을 제안한다.



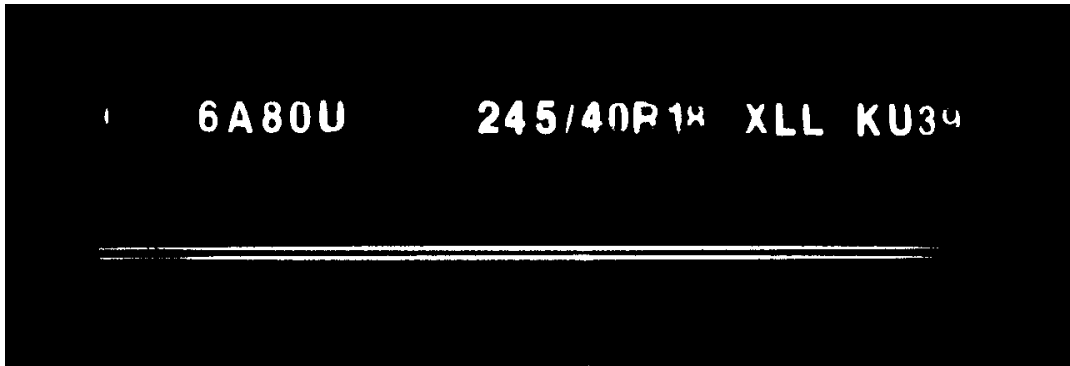
[그림 9] 문자인식 순서도

##### 1) 문자인식을 위한 영상 이진화 방법

영상 이진화는 문자 인식 과정에서 첫 번째로 이루어지는 전처리 과정으로써 입력 받은 컬러 영상을 흑백 영상으로 처리하는 과정이다. 영상 이진화는 그 결과에 따라 문자추출에 많은 영향을 끼치므로 적절한 이진화 방법이 필요하다. 보편적인 이진화 방법은 영상의 초점 또는 명도 대비가 높지 않은 영상에서 이진화를 할 경우 문자와 배경의 구분이 확실하지 않아 배경 영역도 검은색으로 처리될 수 있으며 문자 영역에는 선명하지 않은 결과가 나온다. 본 실험에 사용되는 입력영상은 산업체 현장에서 획득한 영상들로 주변 환경에 많은 영향을 받아 문자와 배경을 정확하게 나누지 못하는 경우가 발생한다. 때문에 보편적인 이진화 방법이 아닌 적응적 이진화 방법을 주로 사용한다[16][17].



[그림 10] 입력 영상



[그림 11] 이진화 영상

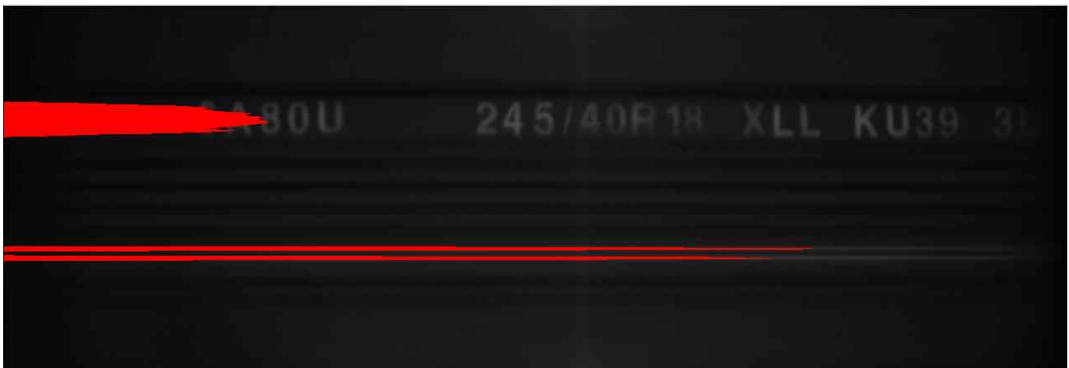
획득한 영상에는 먼지, 공업용 페인트등과 같은 문자와 상관없는 작은 물질도 촬영될 수 있다. 때문에 영상 이진화 처리시 일부 영역에는 작은 노이즈가 다수 발생 할 수 있다. 보다 정확한 문자 추출을 위해 획득 영상에 가우시안 커널을 적용한다. 가우시안 커널을 컨볼루션할 경우 일부 작은 노이즈들을 제거하는 효과를 기대할 수 있다.

## 2) 문자 추출 방법

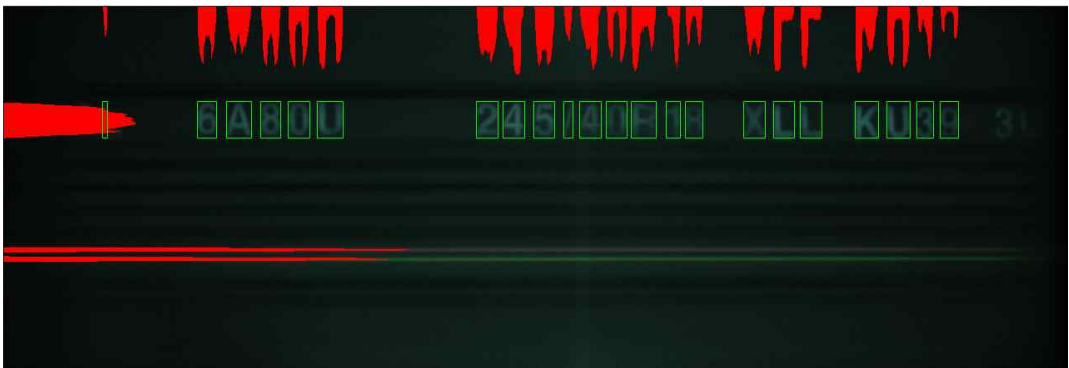
문자 추출은 앞선 단계에서 획득한 이진영상으로부터 문자의 위치를 찾고 하나의 문자씩 분리하여 추출하는 단계이다. 본 논문에서 입력 영상은 수평으로 이동하는 컨베이어 위에 고정된 카메라에서 획득 된 영상으로써 영상의 각도가 변화하지 않음을 가정한다. 앞에서 설명한 가정하에 문자추출은 수평과 수직에 대한 히스토그램 생성을 통해서 이루어진다.



[그림 12] 수직에 대한 히스토그램



[그림 13] 수평에 대한 히스토그램



[그림 14] 문자 추출 결과

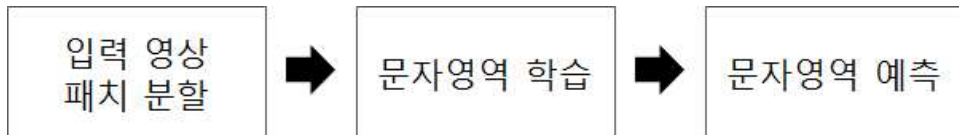
이진 영상으로부터 각 화소의 수직에 대한 이진값들을 누적할 경우 그림 12와 같이

히스토그램이 그려진다. 그림 12의 히스토그램에서 값의 상승시작 위치와 값의 하락 끝 위치로부터 문자의 가로길이를 유추할 수 있다. 이와 마찬가지로 그림 13의 히스토그램에서 값의 상승과 하락 값으로부터 문자의 높이에 대한 정보를 알 수 있다. 최종적으로 수평, 수직에 대한 정보들로부터 각 문자의 위치를 알아낼 수 있다.

하지만 영상 이진화를 통한 문자영역 추측은 주변 환경이 급변할 경우 다량의 노이즈 발생으로 문자 영역의 위치 추적이 어려워진다. 제한된 환경 안에서만 좋은 결과들을 보이는 영상 이진화보다 시시각각 변화하는 환경에서도 적절한 문자 영역을 추출할 수 있는 연구가 필요하다. 본 논문에서는 컨볼루션 뉴럴 네트워크를 사용한 영상 내 문자 위치를 알아내고자 한다.

### 3) 제안하는 문자 위치 추정 방법

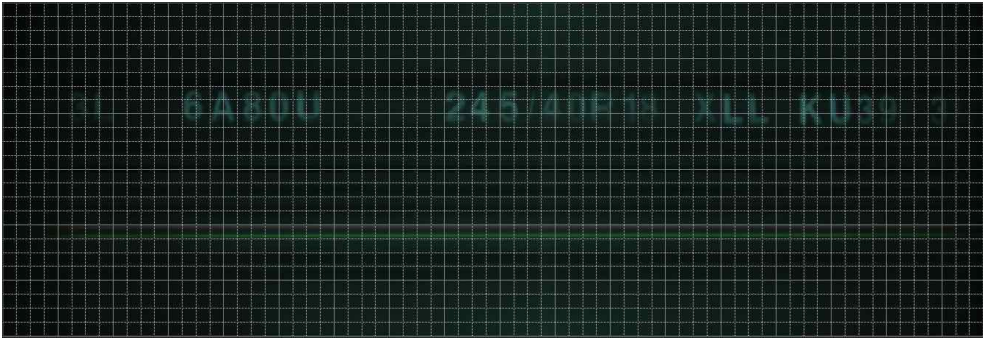
컨볼루션 뉴럴 네트워크를 통한 영상내 문자 위치 추정 방법은 크게 그림 15와 같다. 입력 영상 패치 분할 단계는 입력영상 그림10을 그림 16과 같이 여러 패치로 분할한다. 분할된 각 패치들은 패치 안에 문자가 차지하는 비중을 값을 가지며 컨볼루션 뉴럴 네트워크를 통하여 각 패치들을 학습하게 된다. 학습이 완료된 신경망은 각 패치별 문자영역 비중을 예측하도록 한다.



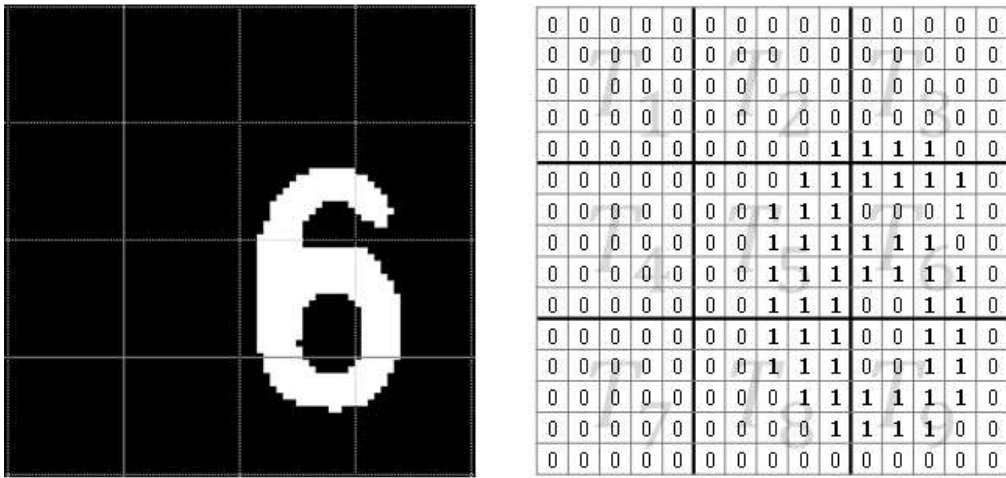
[그림 15] CNN을 통한 문자영역 추측 순서도

각 패치들이 갖게 되는 문자 영역의 비중 값은 그림17과 같이 예를 들 수 있다. 숫자 '6' 주변의 패치모양을 그림 17 (b)와 같이 가로, 세로 3x3의 크기를 가질 경우 T1부터 T9까지 아홉 개의 영역으로 나타 낼 수 있다. 각 영역에 대한 문자영역 비중 값 결정은 식 9와 같이 영역 내의 합연산을 통하여 결정한다. 그림 17(b)에 대한 문자영역 비중 값은 표1과 같다.

$$T_i = \sum_{y=1}^n \sum_{x=1}^m patch(x,y) \quad (\text{단, } 1 \leq i \leq 9) \quad (\text{식 9})$$



[그림 16] 입력영상 패치



(a)

(b)

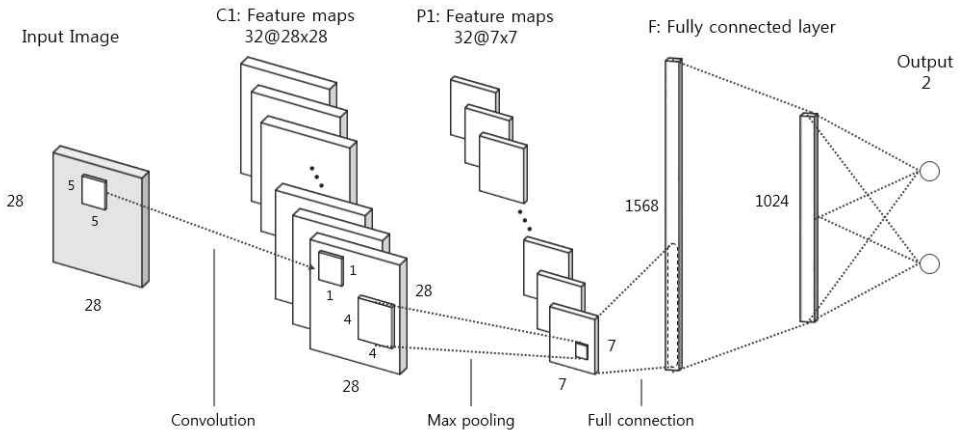
[그림 17] 숫자 6 주변의 패치

$T_1=0$	$T_2=1$	$T_3=3$
$T_4=0$	$T_5=14$	$T_6=11$
$T_7=0$	$T_8=9$	$T_9=11$

[표 1] 패치별 문자영역 추정 값



표1과 같이 구해진 패치별 문자 비중 값을 가지고 컨볼루션 뉴럴 네트워크의 학습을 수행한다. 문자영역 추정을 위한 컨볼루션 뉴럴 네트워크의 구조는 그림 18과 같다.

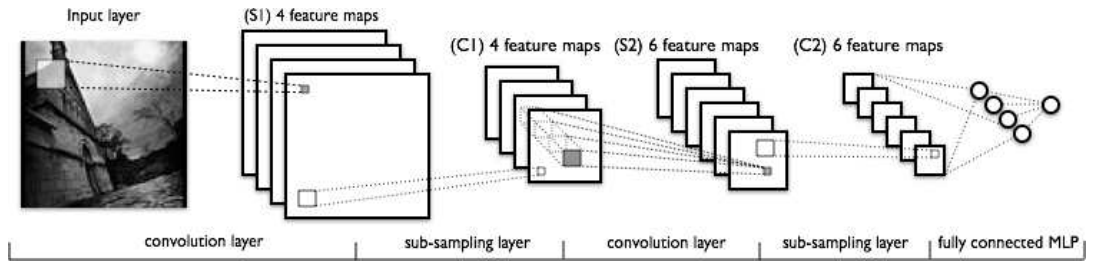


[그림 18] 문자영역 추정을 위한 컨볼루션 뉴럴 네트워크 구조

입력 영상은 전체 입력영상에서 회색조 영상으로 변환된 이미지에서 28x28 크기를 가지는 패치이다. 입력층에서 컨볼루션층인 C1으로는 5x5 크기를 가지는 커널이 5화소의 간격으로 이동하며 컨볼루션 연산이 수행된다. 사용된 커널의 종류는 총 32가지이며 C1층에는 28x28 크기를 가지는 32개의 특징맵이 형성된다. C1층에서 풀링층인 P1층 사이에는 커널 내의 최댓값을 전달하는 max pooling 연산이 수행된다. max pooling에 사용된 커널의 크기는 4x4 크기를 가진다. 전달 받은 P1층에서는 모든 데이터들을 1차원 벡터로 변환하여 1024개의 뉴런을 가지는 은닉층과 완전 연결 형태를 이루게 된다. 최종적으로 출력층은 2개의 뉴런을 가지며 각 뉴런은 문자 영역의 유무를 나타낸다.

### 3) 문자인식 방법

문자인식은 1950년대에 유럽과 미국 등에서 상용화된 광학 문자 인식(OCR)을 발표되며 많은 연구가 이루어졌다. 2000년대부터의 문자인식 기술은 신경망을 이용한 문자인식 성능이 99%에 이르는 높은 인식률을 보이며 신경망 사용이 문자인식의 주된 기술로 자리 잡았다.



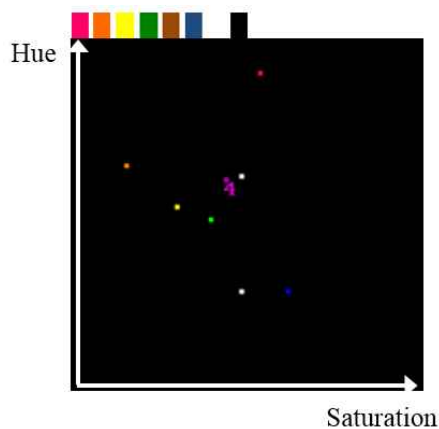
[그림 19] LeCun의 Convolution neural Network 구조[18].

문자인식에 사용된 모델은 그림 19와 같다. 그 구성은 2개의 컨볼루션층 C1과 C2, 2개의 풀링층 S1과 S2 그리고 완전연결형태의 네트워크 층과 출력층으로 구성된다.

## 2. 색상 인식

### 1) 색 분류

기존에 사용하였던 색 분류 방법은 획득한 색선의 입력영상으로부터 임의의 대표 픽셀을 추출한다. 추출된 대표픽셀은 그림20과 같이 채도와 명도로 표현된 2차원 HSV 색공간에 가이드된 고유색의 위치와 분류하고자 하는 대표 픽셀의 색상 위치와 비교하여 유클리디안 거리가 가장 가까운 고유색으로 분류하는 방법을 사용하였다.



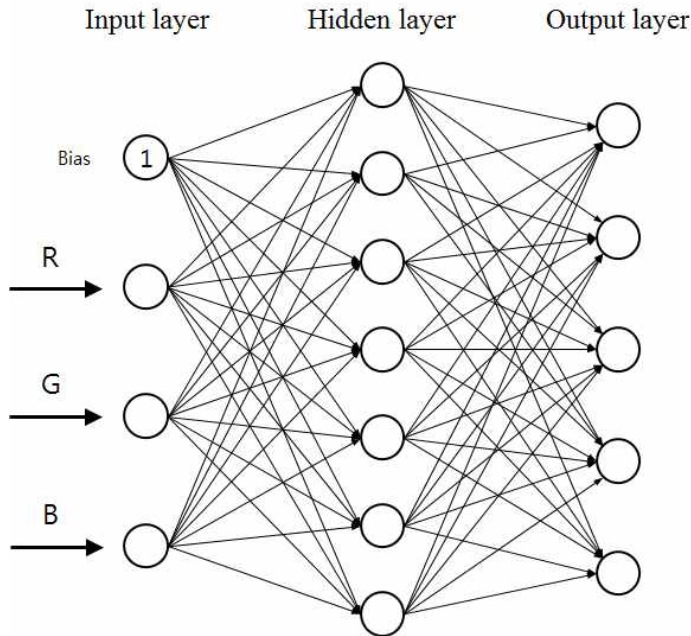
[그림 20] 명도와 채도로 표현된 2차원 색공간



[그림 21] 색선 확대 영상과 그 구성 색상들

하지만 본 방법은 임의의 위치에 추출된 픽셀들이 항상 동일한 색상을 가지고 있지 않으며 주변 환경이나 도료의 농도에 따라 다르다. 그 예로 그림 21과 같이 같은 색선 위에 추출한 화소이지만 다양한 색상이 구성된 것을 볼 수 있다. 때문에 소수의 대표 화소로 전체 색선의 색상을 분류하는 방법은 환경의 영향을 많이 받는다.

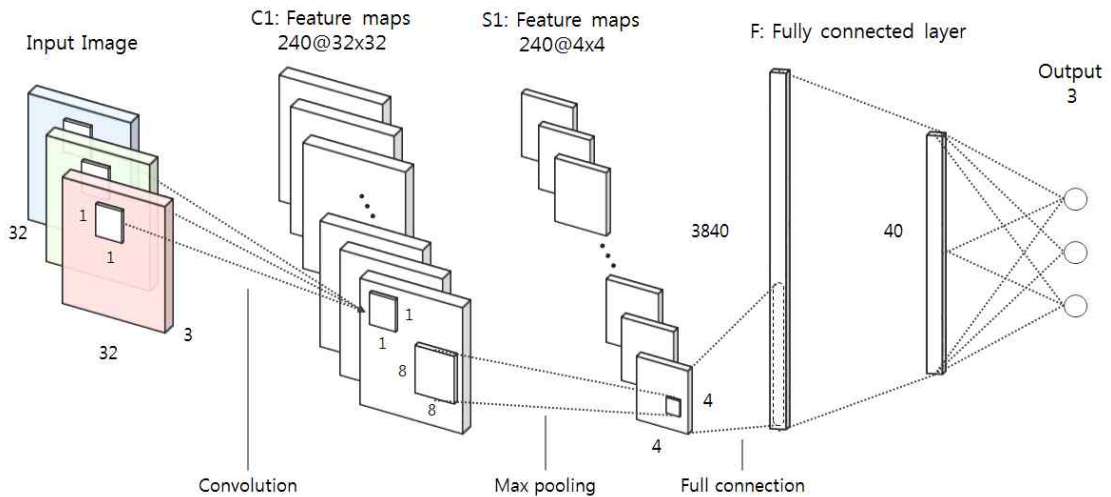
본 논문에서는 소수의 대표 화소가 아닌 색선 위의 다수의 화소에 대한 정보에 대하여 학습을 통해 색을 분류하고자 한다. 색상인식을 위한 색 분류 방법은 그림 22와 같이 전방향 뉴럴네트워크를 사용하여 분류한다.



[그림 22] 전방향 뉴럴네트워크

## 2) 제안하는 광원추정 Convolution Neural Network 구조

본 논문에서 제안하는 Convolution Neural Network의 구조는 그림 23와 같다. 네트워크는 하나의 컨볼루션층과 하나의 풀링층 그리고 완전 연결된 전방향 신경망으로 구성된다. 입력 영상의 크기는 32x32 크기로 재구성 한다. 컨볼루션층을 나타내는 C1은 입력 영상의 RGB 각 채널에 대하여 1x1x3 크기의 커널을 컨볼루션한 결과를 전달받는다. 일반적으로 이미지 분류에 사용되는 CNN 커널들은 영상내의 다양한 밝기 변화를 찾기 위해 3x3 이상의 크기를 갖지만 본 논문의 CNN에 사용되는 커널은 각 화소가 가지는 R, G, B 세 채널간의 관계를 찾기 위하여 1x1x3 형태의 커널을 사용한다. 사용한 커널의 수는 240개이며 240개의 특징맵을 생성하게 된다. 풀링층을 나타내는 S1은 8x8 크기의 커널과 8화소 간격으로 C1의 각 특징 맵 안을 이동하며 Max pooling 된 결과를 전달받는다. C1의 Max pooling 출력결과는 3x3 크기의 240개 특징맵을 갖는다. 완전 연결층은 4x4x240 크기의 S1층의 데이터를 재구성하여 3,840개의 벡터 형태로 가지게 된다. 완전 연결층은 40개의 뉴런을 가지는 Hidden층과 완전연결 형태로 연결된다. 최종적으로 출력층에는 3개의 뉴런으로 구성되며 각각의 뉴런은 광원을 표현하는 R, G, B로 나타낸다.

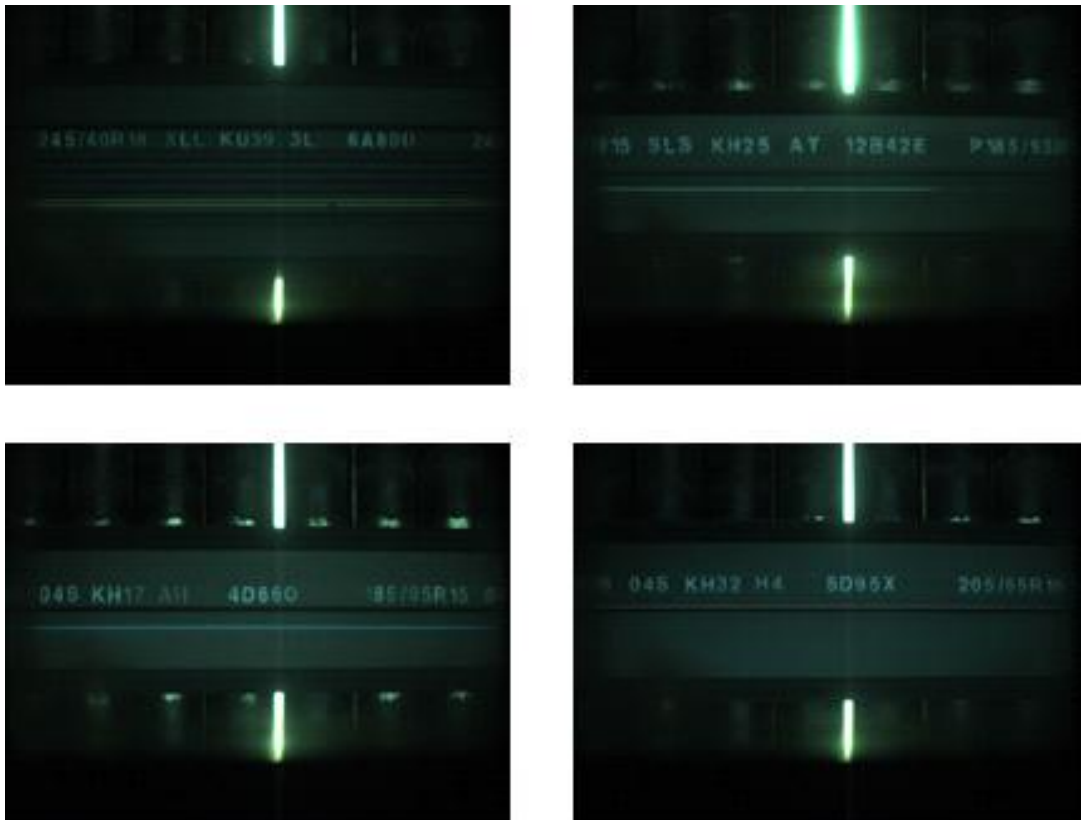


[그림 23] 광원 추정을 위한 컨볼루션 뉴럴 네트워크 구조

## IV. 실험 및 평가

### 1. 실험 데이터

본 논문에서 사용된 영상 데이터는 그림24와 같이 산업체에서 획득한 타이어 영상을 사용한다. 각 획득 영상은 1278x958 크기를 가지며 총 150장의 이미지를 사용한다. 그리고 광원 추정을 위한 컨볼루션 뉴럴 네트워크의 학습율을 알아보기 위하여 그림 26, 27과 학습 및 테스트 영상에 대하여 실험을 한다. 사용된 데이터 수는 학습 및 테스트 각각 900개의 영상으로 이루어져 있다.



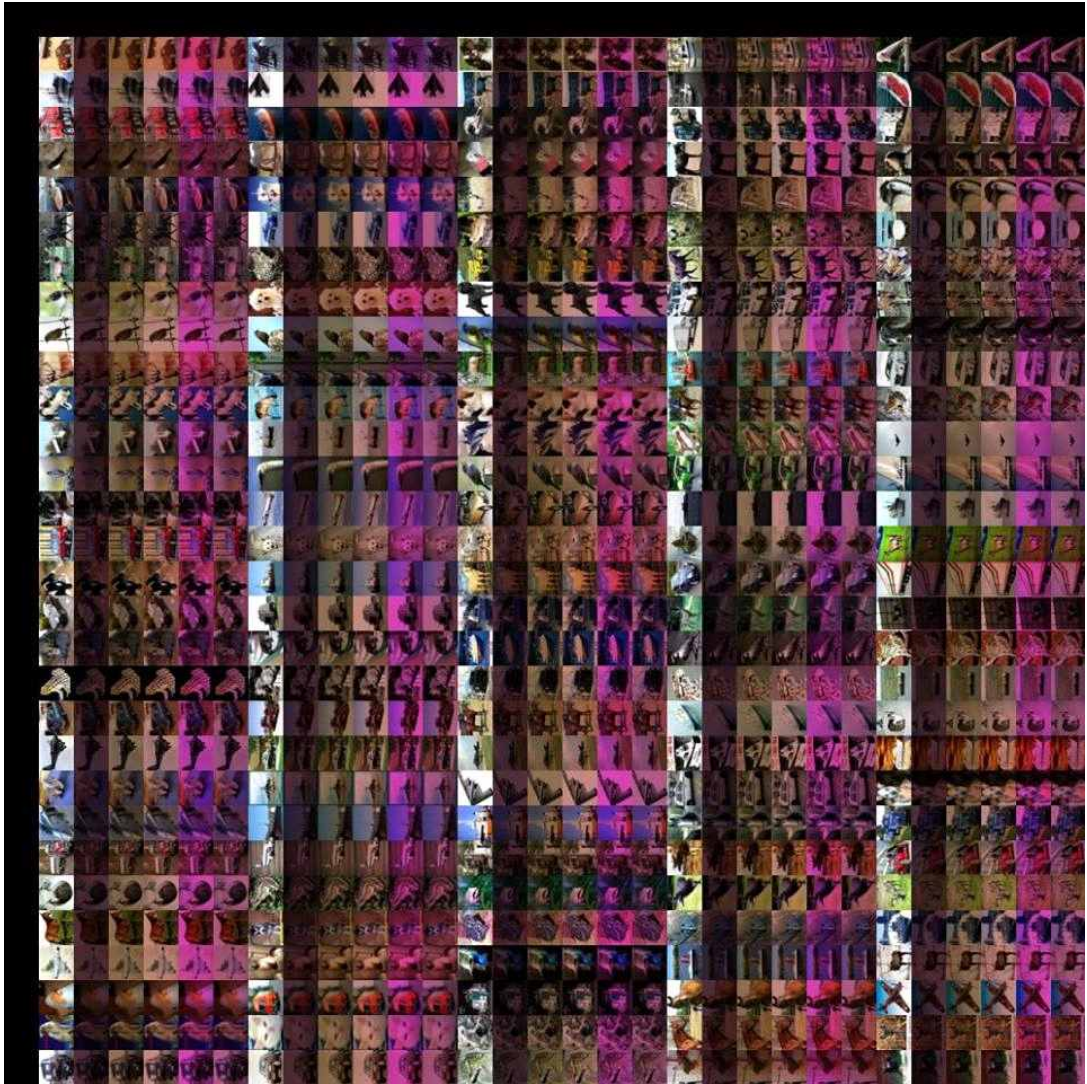
[그림 24] 획득 영상



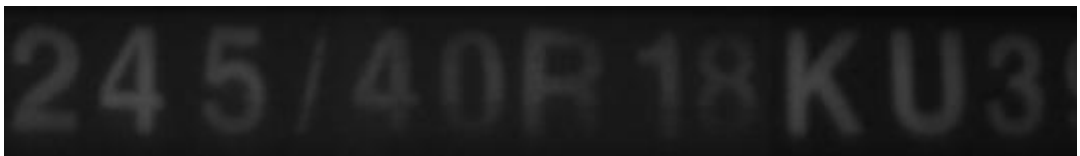
[그림 25] 문자영역



[그림 26] 광원 추정 학습 데이터



[그림 27] 광원 추정 테스트 데이터

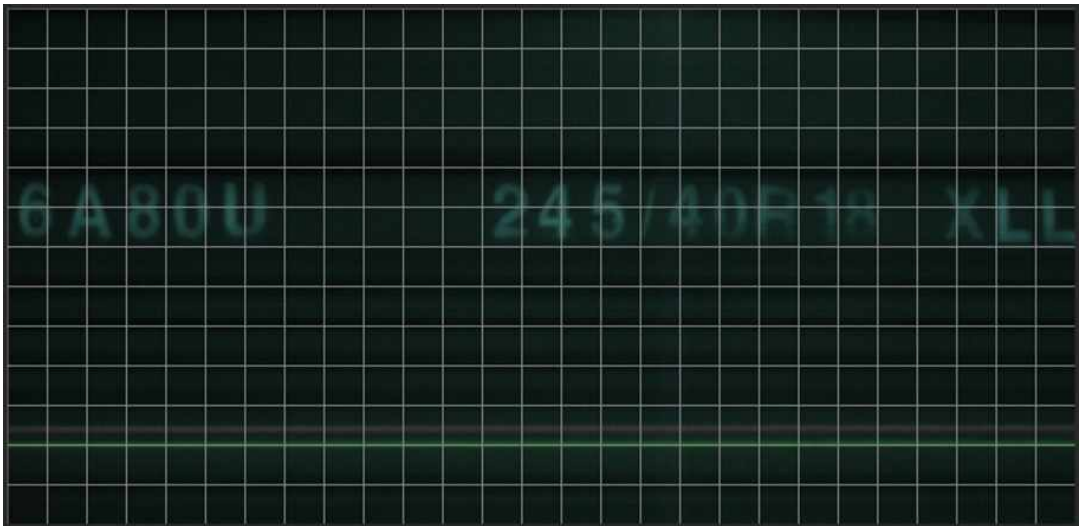


[그림 30] 문자영역 학습데이터

## 2. 실험 방법

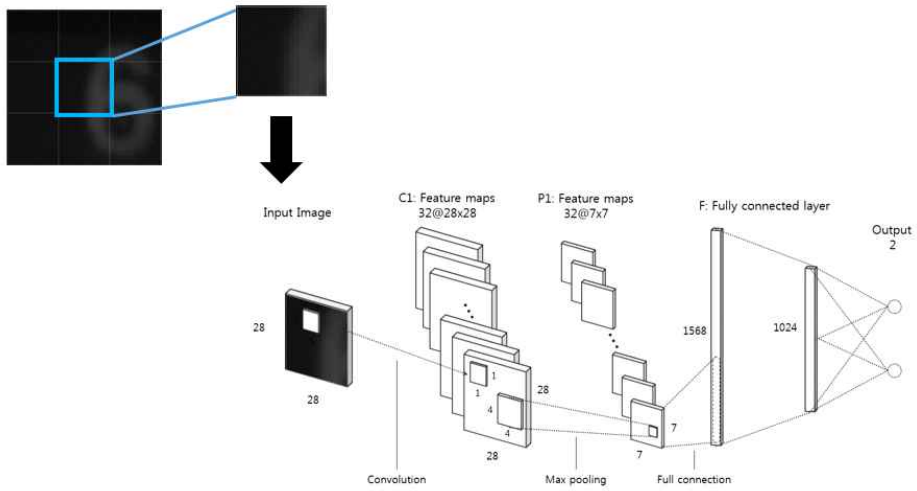
### 1) 문자인식

문자인식 실험은 크게 학습과 테스트 두 단계로 나뉘어 진행된다. 학습 단계에서는 추출된 타이어 영역 이미지의 패치영역을 컨볼루션 뉴럴 네트워크를 통하여 학습한다. 학습에 사용되는 패치 영역은 그림 31과 같다. 패치 이미지는 타이어 영역 전체 이미지를 28x28 크기로 분할된 영역을 나타낸다. 학습에 사용된 컨볼루션 뉴럴 네트워크 구조는 앞장에서 설명한 것처럼 그림 32와 같은 한 개의 컨볼루션층과 한 개의 풀링층 그리고 완전연결 형태 및 출력층으로 구성된다. 출력층은 2개의 뉴런으로 구성되며 One-Hot Encoding 방법에 따라 0 또는 1로 패치 영상이 문자영역인지의 유무를 나타내도록 한다. 학습 알고리즘은 경사하강법 및 오류 역전파 알고리즘을 따른다. 학습 단계가 완료되면 테스트를 수행하며 테스트에 사용되는 이미지는 그림 33과 같다. 학습된 네트워크층을 사용하여 영상의 문자영역을 나타내도록 한다.



[그림 31] 학습 입력 영상





[그림 32] 패치영상 과 CNN 실험



[그림 33] 테스트 입력 영상

## 2) 색상인식

색상인식 실험은 우선 입력 영상의 전체 광원을 추정하기 위하여 제안된 색상인식 컨볼루션 뉴럴 네트워크를 사용하고 타겟 광원 값과 결과를 비교하는 방법으로 진행된다. 또한 색선에서 추출된 화소의 색상분류는 전방향 뉴럴 네트워크를 사용하여 그 결과를 확인한다.

### 3. 실험 결과

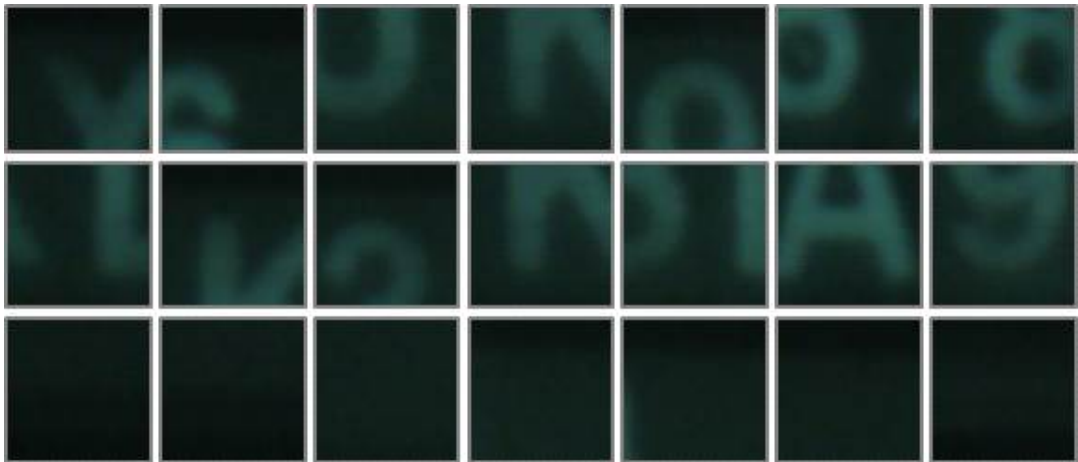
문자 영역 위치 추정을 위한 컨볼루션 뉴럴 네트워크 실험은 실험 방법에서 설명한 바와 같이 실험된다. 실험환경은 표 2와 같다.

구 분	품 명	비 고
OS	ubuntu 14.04 LTS	
CPU	Intel Core i7-4790	3.60GHz
RAM	Samsung DDR3 8GB	
Language	Python2.7	
Deep Learning Library	TensorFlow[19]	

[표 2] 실험 환경

#### 1) 문자 영역 추정 실험

패치 크기는 28x28로 총 28개의 패치 영역으로 학습된다. 학습된 주요 영상은 그림 34와 같다. 학습은 8000회의 반복 수행하며 학습이 이루어졌으며 학습율은 0.0001로 설정하였다. 학습 시 loss 값은 0.3316이 나왔다. 테스트 결과 90% 정확성을 보였으며 예외적인 이미지는 그림 35와 같다.



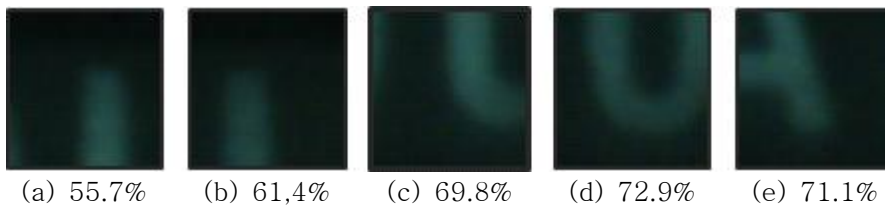
[그림 34] 학습 영상 패치

학습 패치 수	28
반복 수	10000
학습율	0.0001
학습 정확성	90%
테스트 정확성	80%

[표 3] 학습 결과



[그림 35] 오류가 발생 된 이미지

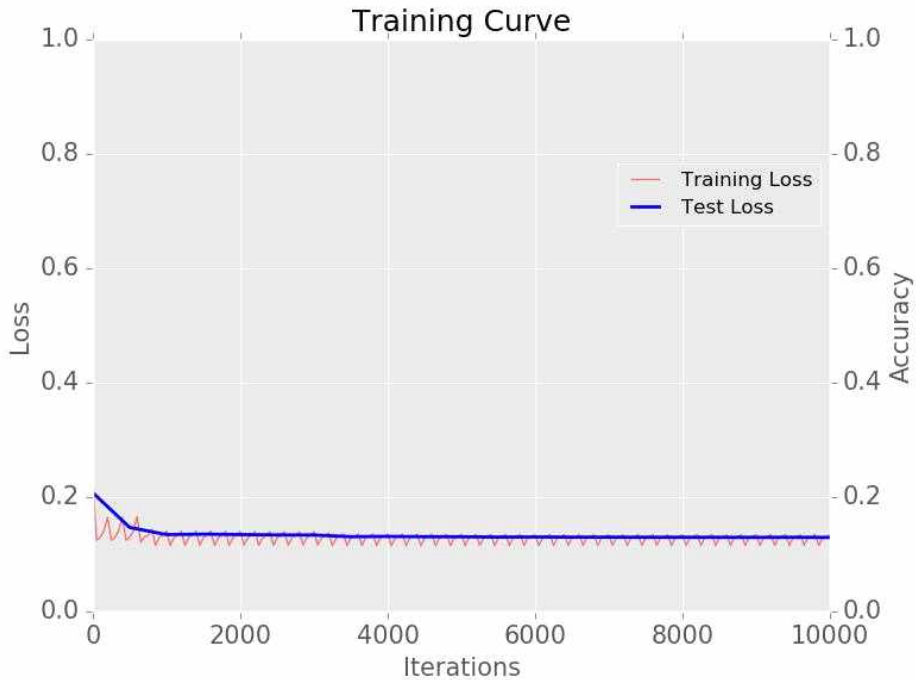


[그림 36] 문자 영역 예측 결과

테스트 결과 그림 36과 같이 패치 내 문자 영역이 차지하는 비중에 따라 예측 값이  
 다름을 확인 할 수 있다. 오류가 발생된 데이터는 그림 35와 같이 문자영역이 20% 미  
 만으로 차지한 경우에 해당한다. 작은 영역에 대한 정확성을 높이기 위해서는 해당 영  
 역들에 대한 학습 데이터 셋을 추가적으로 모을 필요성이 있다.

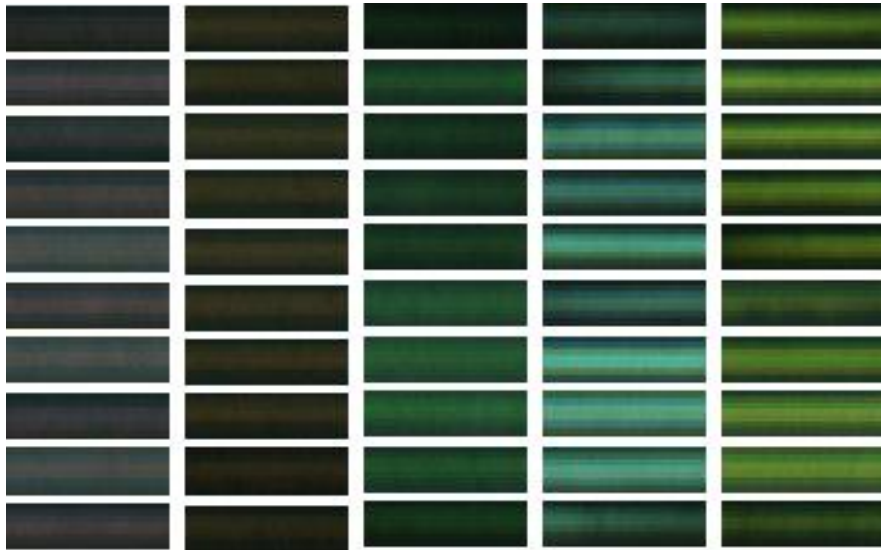
## 2) 광원 추정 및 색상 분류 실험

광원 추정을 위한 컨볼루션 뉴럴 네트워크 실험에서는 그림 26의 학습 데이터의 광원 타겟 값을 설정하여 각 영상별 네트워크 학습을 시도하였다. 반복수는 10,000회 반복하며 학습 커브는 그림 37과 같다. 최종 loss 값은 0.1342 이며 테스트 결과 87.1%의 성능을 나타냈다.

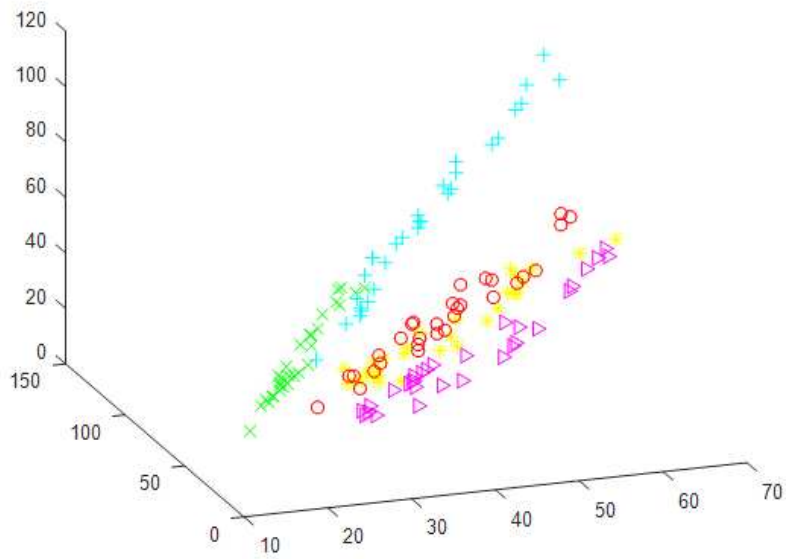


[그림 37] 광원 추정 컨볼루션 뉴럴 네트워크 학습 커브

전방향 신경망을 통한 색상 분류에서는 그림38과 같이 120x1000 크기의 5가지 색선 으로부터 색상 영역을 추출하여 학습 및 테스트가 이루어졌다. 추출된 색선들의 데이터 분포는 표4 그리고 그림 39와 같다. 사용된 학습 데이터 수는 104, 확인 데이터 23, 테스트 데이터 23개 등 총 150개의 데이터로 이루어졌다. 실험결과 그림 40과 같이 모든 색선에 대한 예측결과가 100%로 우수한 성능을 나타냈다.



[그림 38] 확대된 색선 영상



[그림 39] 색상 데이터 분포 그래프

	R	G	B
적색	6315903	6642637	6575019
	5022774	5825463	5809225
	6926284	8452349	8148142
	6892354	8136428	7775217
	7051880	8379678	7981152
초록	3123498	6820641	4258015
	2257710	4892804	2816005
	4061756	8992795	5483655
	4191143	8957249	5617217
	3719587	8185879	5359224
청록	6980376	13322211	10860031
	7752146	14372368	11482767
	7223651	13954871	11589613
	7606083	14905033	12458535
	4346325	7970233	7035426
노랑	8244168	12118658	5273844
	7648472	11527483	5062052
	6550650	10209461	4184622
	4780112	7534369	4062871
	4360111	6402529	2498839
갈색	3433015	3521073	2369305
	4282501	4439129	3004141
	4286247	4648747	3168478
	4446560	4934118	3355029
	4628079	5196202	3512538

[표 4] 색상 데이터 분포



[그림 40] Confusion Matrix

## V. 결 론

본 논문에서는 딥러닝을 통한 문자 영역 추정 및 광원 추정 등 문자 및 색상 인식에 필요한 전처리 과정을 딥러닝을 통하여 해결하며 신경망을 통한 색상분류를 하고자 한다. 따라서 본 논문에서는 다양한 환경변화에도 강인한 영상 내의 문자 영역의 추정과 광원 추정을 위하여 딥러닝을 통해 시도하였다. 딥러닝을 통한 문자 영역 추정은 입력 영상의 모든 패치들에 대하여 문자 영역과 배경 영역 두 그룹으로 나누어 영상 패치 분류 방법을 제안하였으며 광원 추정 또한 지역적 광원을 추정하기 위하여 패치별 광원 추정을 하였다. 사용된 컨볼루션 뉴럴 네트워크 구조는 하나의 컨볼루션층과 하나의 풀링층을 가지는 구조를 가진다.

문자 영역 추정 실험에는 제안된 컨볼루션 뉴럴 네트워크 구조를 통하여 문자영역과 비 문자영역에 대한 학습을 통하여 입력 영상의 패치별 문자영역 예측 실험을 하였다. 실험결과 80%의 정확성을 보였으며 패치의 문자영역 비중에 따라 예측률이 달라짐을 확인 할 수 있다.

광원 추정에는 실험 결과 타겟 광원 RGB값과 예측된 RGB값 최대 20의 차이를 보였으며 전체 정확성은 87.1%의 성능을 보인다.

색상 분류 실험은 평균 색상을 구하고 전방향 신경망 구조를 통하여 실험한 결과 100%의 분류 성능을 보여줬다.

향후에는 문자 영역 추정에 및 광원 추정에 사용된 컨볼루션 뉴럴 네트워크 구조의 컨볼루션층 및 풀링층 수 증가 시키는 등 구조 변형을 시도하여 보다 효율적으로 변형하여 학습을 향상과 오류율을 최소화 하는 연구가 필요하다.



## 【참고문헌】

- [1] G. Buchsbaum, "A Spatial Processor Model for Object Colour Perception," J. Frankl. Inst. Vol. 310, No. 1, pp. 1-26, 1980.
- [2] K. Barnard, "Practical Colour Constancy," Diss. Simon Fraser University, 1999.
- [3] G.D. Finlayson, D.H. Steven, and M.H. Paul, "Color by Correlation: A Simple, Unifying Framework for Color Constancy," IEEE Trans. on Pattern Anal. Mach. Intell. Vol. 23, No. 11, pp. 1209-1221, 2001.
- [4] G. Finlayson, S. Hordley, I. Tastl, "Gamut Constrained Illuminant Estimation," Int. J. Comput. Vis., Vol. 67, No. 1, pp. 93-109, 2006.
- [5] W. Xiong, B. Funt, "Estimating Illumination Chromaticity Via Support Vector Regression," J. Imag. Sci. Technol., Vol. 50, No. 4, pp. 341-348, 2006.
- [6] H. Joze and M. Drew, "Exemplar-Based Color Constancy and Multiple Illumination," IEEE TPAMI, Vol. 36, No. 5, pp. 860-873, 2014.
- [7] P. Gehler, C. Rother, A. Blake, T. Minka, and T. Sharp, "Bayesian Color Constancy Revisited," Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-8, 2008.
- [8] V. Cardei, B. Funt, K. Barnard, "Estimating The Scene Illumination Chromaticity Using a Neural Network," J. Opt. Soc. Amer. A, Vol. 19, No. 12, pp. 2374-2386, 2002.
- [9] R. Stanikunas, H. Vaitkevicius, and J.J. Kulikowski, "Investigation of Color Constancy with a Neural Network," Neural Networks, Vol. 17, pp. 327-337, 2004.
- [10] Y. LeCun, Y. Bengio, G. Hinton, "Deep Learning," Nature, Vol. 521, pp. 436-444, 2015.
- [11] J. Schmidhuber, "Deep learning in Neural Networks: An overview," Neural Networks, Vol. 61, pp. 85-117, 2015.
- [12] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, "Gradient-Based Learning Applied to Document Recognition," Proc. IEEE, Vol. 86, No. 11, pp. 2278-2324, 1998.
- [13] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet Classification with Deep Convolutional Neural Networks," In Advances in Neural Information Processing Systems 25, pp. 1106 - 1114, 2012.

- [14] Z. Lou, T. Gevers, N. Hu, M. Lucassen, "Color Constancy by Deep Learning," in: Proceedings of British Machine Vision Conference, 2015.
- [15] McCulloch, Warren S., and Walter Pitts, "A Logical Calculus of The Ideas Immanent in Nervous Activity," The bulletin of mathematical biophysics, Vol. 5, issue 4, 1943, pp. 115-133.
- [16] J. Ohya, A. Shio, S. Akamatsu, "Recognizing Characters in Scene Images," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 16, No. 2, pp. 214-220, 1994.
- [17] R.L. Hoffman and J.W. McCullough, "Segmentation Methods for Recognition of Machine-Printed Characters," IBM I. Research and Development, pp. 153-65, 1971.
- [18] <http://deeplearning.net/tutorial/lenet.html>
- [19] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Good-fellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems," arXiv 1603.04467, 2016.