

# Development of Analog Gauge Recognition System Using Morphological Operation and Periodic Measurement Function

Jin-kyu Ryu\*, Young-Tae Kwak\*\*

## Abstract

In this paper, we propose a new method to read the hand of analog gauges to prepare for the smart factory. In addition, we suggest a new and improved method that can apply, in general, diverse analog gauges even if their scale types and ranges are various. Many companies are making great efforts to build smart factories that increase energy efficiency and automation. Managers use a variety of equipment and tools to manage the production process at the factory. In this kind of factory, analog gauges have been often used with many equipment and tools. Analog gauges are mostly circular in shape, and most papers use circular hough transform to find the center and radius of a circle. However, when the object to be found is not of the correct circle type, it takes a long time to recognize the circle using the circular hough transform, and the center and radius of the circle can not be calculated accurately. The proposed method was tested on various circular analog gauges. As a result, we confirmed that our method is outstanding.

▶ Keyword: Analog Gauge Recognition, Morphological Operation, Distance Matching Function

## 1. Introduction

최근 많은 기업이 4차 산업혁명의 대표 중의 하나인 스마트 팩토리를 구축하기 위해서 많은 노력을 하고 있다[1]. 스마트 팩토리는 말 그대로 ‘똑똑한 공장’이라는 뜻으로 IT 기술을 사용하여 에너지 효율을 높이고, 자동화 비중을 높인 공장을 의미한다.

스마트팩토리에서는 여러 장비에 각각의 다양한 센서들이 부착되어 있고, 인터넷을 기반으로 모든 장비를 연결하여 장비와 장비, 사람과 장비 간의 데이터를 공유하게 된다[2]. 관리자는 직접 현장에 가서 확인하지 않고, 스마트폰이나 컴퓨터에서 전송되는 데이터를 가지고 모니터링 및 제어할 수 있다.

공장에서는 생산공정을 관리하기 위해 여러 장비와 기구를 사용한다. 아날로그 게이지는 압력과 온도, 전압 등을 측정하는 대표적인 장비로써, 쓰이는 용도에 따라 눈금의 표시 방법과 모양도 매우 다양하다[3]. 스마트 팩토리의 관제 서비스에서는 주로 카메라로 촬영한 영상을 이용해서 실시간으로 바늘이 가

리키는 눈금의 값을 취득하게 된다[4].

아날로그 게이지는 대부분 원형 형태를 갖추고 있으며, 원형 형태를 인식하는 방법으로 원형검출 허프변환을 이용한다[5]. 그러나 원형검출 허프변환은 이미지의 크기가 커질수록 원을 찾는 데 필요한 자원이 많이 필요하고, 카메라 각도 문제 및 게이지의 형태 자체가 정확한 원형 형태가 아닐 경우 원의 중심과 반지름을 정확하게 구할 수 없게 된다.

본 논문에서는 원을 인식하기 위해 사용하는 원형검출 허프 변환방법 대신 객체의 크기 비교를 통해서 게이지의 중심부를 인식하고, 아날로그 게이지에서 일정하게 나타나 있는 눈금의 주기성을 활용하여 눈금의 범위를 찾아 바늘이 가리키는 값을 측정하는 방법에 대해 제안한다.

논문의 전개는 다음과 같다. 2장에서는 기존 아날로그 게이지 측정 방법에 대하여 고찰하고, 각 방법의 한계와 문제점을 설명한

---

• First Author: Jin-kyu Ryu, Corresponding Author: Young-Tae Kwak  
\*Jin-kyu Ryu (lightyu@jbnu.ac.kr), Dept. of Computer Engineering, Chonbuk National University  
\*\*Young-Tae Kwak (ytkwak@jbnu.ac.kr), Dept. of IT and Engineering, Chonbuk National University  
• Received: 2017. 11. 25, Revised: 2017. 12. 28, Accepted: 2018. 02. 01.

다. 그리고 본 연구와 관련된 기존 연구를 살펴볼 것이다. 3장에서는 아날로그 게이지를 인식하는 방법, 영상에서 원의 중심을 추출하는 방법, 주기성을 이용한 패턴을 찾는 법을 소개한다. 4장 실험에서는 여러 아날로그 게이지를 대상으로 눈금 범위를 구하고 바늘이 가리키는 게이지 값을 출력하고 실제 이미지에서 바늘이 가리키는 값과 비교해서 아날로그 게이지 인식에 관한 성능을 확인한다. 그리고 마지막으로 결론을 맺는다.

## II. Related Works

Gellaboina는 원형검출 허프변환을 통해 아날로그 게이지의 영역을 감지하고, 이미지에서 아날로그 게이지를 따로 분리한다[6]. 분리된 이미지를 원형검출 허프변환을 통해 구한 원의 중심과 반지름을 극좌표를 이용하여 선형 스케일로 변환한다. 일반적인 아날로그 게이지의 바늘은 길고 검은색의 특징을 가진다. 변환된 이미지의 열 투영을 분석하여 가장 큰 값에 해당하는 인덱스를 바늘의 위치로 판단한다.

이러한 방법은 하나의 아날로그 게이지에 대해 같은 매개변수를 적용하며, 원형 형태를 인식할 때 일반적으로 원형검출 허프변환을 사용하고 있다. 그러나, 아날로그 게이지는 용도에 따라서 형태가 매우 다양하며, 원이 아닌 타원 형태를 인식할 때 계산 시간이 증가하게 되고, 원의 중심을 잘못 인식하는 문제가 발생해서 눈금과 게이지 값에 오차가 발생할 수 있다.

Danilo는 전압이나 전류 측정에 쓰이는 부채꼴 모양의 아날로그 게이지를 대상으로 연구를 진행하였다[7]. 직선 허프변환을 사용하여 현재 게이지 값을 가리키는 바늘의 인덱스 값을 취득하고, 중심에서부터 바늘이 가리키는 처음과 끝 눈금 사이의 각도를 계산해서 각도와 눈금의 비율에 따른 게이지 값을 알아내는 방법을 제안하였다.

실험은 부채꼴 모양의 게이지를 가지고 테스트를 진행하였다. 게이지를 인식하기 위해서 바늘과 눈금 사이의 각도가 필요하기 때문에 게이지의 중심과 최소 눈금, 최대 눈금의 각도가 0° 이상이고, 준비 과정에서 바늘이 시작 눈금에 있는 각도와 최대 눈금에 있는 각도를 알고 있어야 한다. 또한, 아날로그 게이지 안에 상표 혹은 기타 문구로 인하여 노이즈가 발생하면 바늘 인식에 어려움이 있다.

Danilo와 마찬가지로 부채꼴 모양의 아날로그 게이지를 대상으로 연구를 진행한 Alegria는 이미지를 기하학적 변환을 통해서 곡선 형태로 이루어진 눈금 선을 수평선에 맞게 정렬하고, 바늘도 수직으로 정렬되는 방법을 제안하였다[8]. 변환된 이미지는 허프변환을 통해서 구한 가장 긴 선분이 현재 게이지 값을 가리키는 바늘 영역이고, 눈금 영역은 푸리에 변환을 통해 가장 큰 주기성을 갖는 행에 해당한다. 게이지 값은 눈금 범위의 수평좌표에서 원본 이미지의 수평선과 일치하는 각도의 비율을 비교하여 계산하였다.

Alegria의 방법은 게이지 안에 있는 눈금이 규칙적이므로 주기성을 활용하여 눈금의 범위와 바늘을 인식하는 방법에 대해 말하고 있다. 그러나 게이지의 눈금 영역 형태가 부채꼴이 아닌 원형으로 돌린 형태일 경우에 게이지의 눈금이 한 줄에 이어져 있지 않고, 분리되어 나타나기 때문에 눈금의 범위를 제대로 인식하지 못할 수 있다.

두 바늘의 각도 차를 이용한 다른 방법으로 제안한 Xiaofeng는 게이지의 바늘이 최소 눈금을 가리키는 이미지와 바늘이 현재의 게이지 값에 해당하는 눈금에 있는 이미지를 비교하여 두 바늘 사이의 각도 차를 이용해서 게이지 값을 구하는 원형 정합법을 사용하였다[9]. 바늘을 인식하기 위해서 허프변환을 사용하고, 이미지를 얇게 작업하는 Image Thinning을 시행하여 바늘이 가리키는 눈금의 정확도를 향상시켰다.

Asha는 어떤 이미지에서 반복적인 패턴을 찾기 위해서 푸리에 변환, Autocorrelation, Reney's entropy, Co-occurrence matrix 속성 등과 같은 주기성을 찾는 여러 방법 중에서 Distance Matching Function과 Haar wavelet을 이용하여 이미지의 행 또는 열 방향에 따라서 주기를 검출하는 방법을 제안하였다[10]. 섬유 이미지, 체스보드와 같은 반복 패턴에 대해서 주기성을 검출하는 연구를 진행하였으며 이미지 합성, 컴퓨터 비전, 텍스처 합성 및 압축과 같은 다양한 그래픽 분야에서 유용할 것이라고 말하고 있다.

## III. The Proposed Method

본 논문에서는 아날로그 게이지의 중심점과 바늘을 인식하기 위해서 원형검출 허프변환(Circular Hough Transform)을 이용한 방법이 아닌 형태학적 처리 방법인 열립과 침식 연산을 통하여 중심점을 구하고 이미지 안에서의 주기성을 이용하여 눈금의 범위와 그리고 최종적으로 게이지의 바늘이 현재 가리키고 있는 값을 출력하는 알고리즘을 제안한다.

전체적인 프로그램 구성도는 그림 1과 같다. 그림 1은 원형 형태의 아날로그 게이지가 포함된 영상으로부터 바늘이 가리키고 있는 값을 출력하기까지의 각 단계를 보여준다.

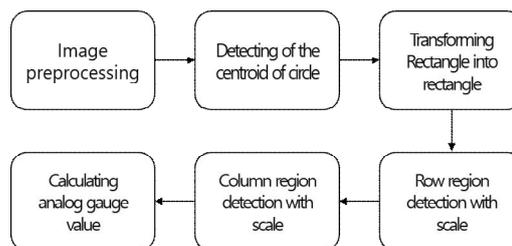


Fig. 1. Process of analog gauge recognition

### 3.1 Image Preprocessing

어떤 종류의 디바이스로 촬영한 영상의 경우에는 이미지가 회전되어 저장되는 문제가 발생하게 된다. 따라서 영상의 정보를 먼저 읽어 와서 회전 정보를 확인한 다음, 본 논문에서 필요로 하는 이미지로 회전한다. 카메라로 촬영 시, 일반적으로 사진은 직사각형의 비율로 촬영된다. 아날로그 게이지는 원형 형태이며, 정사각형 모양의 형태를 갖춘 이미지를 사용하는 것이 중심점을 찾거나 눈금 영역을 찾을 때 편리하다.

카메라를 이용하여 아날로그 게이지를 직사각형의 중심 부분에 위치하도록 촬영한다. 촬영된 이미지의 중심을 기준으로 가로/세로 자르기(clipping)를 구현한다. 즉, 직사각형 영상을 정사각형 영상으로 변환한다. 그리고 원형 게이지의 3×3의 분할 영상 중, 원형의 중심점 추출을 위하여 (2, 2) 영상을 선택한다. 그림 2는 이미지 전처리를 통해 중심점 영역을 추출하는 과정을 나타낸 것이다.

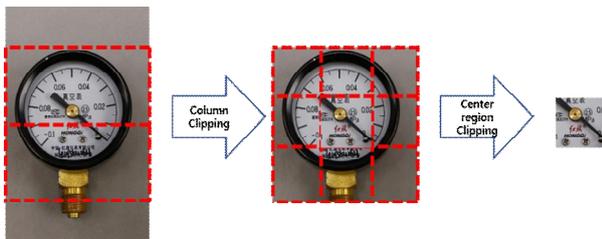


Fig. 2. Image preprocessing for center area cropping

### 3.2 Detecting of the centroid of circle

이미지 전처리 작업을 통해서 얻은 이미지에는 원형 형태의 아날로그 이미지를 포함한다. 원형 형태의 이미지의 중심점을 찾는 여러 방법이 있다. 그중에서 일반적으로 원형검출 허프 변환을 이용한다. 그러나 원형검출 허프 변환을 사용할 경우 많은 연산을 해야 하고, 게이지의 형태가 정확한 원이 아닐 때 발생하는 오차로 인하여 정확한 중심점을 찾는 것이 어려워진다.

본 논문에서는 중심점을 찾기 위해서 그레이스케일로 변환된 이미지를 전체적인 픽셀값의 평균 레벨에 맞추어서 Otsu's 방법으로 이진화한다. 이진화 영상에서 가장 큰 객체를 선택한 후, 구멍(hole)을 메워 내부가 채워지게 되고 중심점이 포함되어 있는 객체가 생성된다. 생성된 객체의 면적을 구한 다음 그 면적이 이진화된 이미지 크기의 일정 부분의 크기가 될 때까지 열림 연산과 침식 연산을 반복한다. 여기서 본 논문은 이진화된 이미지 크기의 1/49로 정하였다.

형태학적 연산에서 대상 영역이 좁아지는 침식 연산과 대상 영역에서 세부 영역이 제거되는 열림 연산을 반복적으로 시행한다. 중심점에 해당하는 부분이 가장 큰 면적에 해당하기 때문에 그 이외의 면적은 반복적인 연산을 통해서 제거한다. 제거되고 남은 1개의 최종 객체에 대해서 중심점을 계산한다.

열림과 침식의 반복 과정에서 중심점을 포함한 객체의 수가 1개일 경우에는 그 객체를 선택하여 열림과 침식을 반복한다. 객체

의 수가 2개 이상일 경우에는 면적이 가장 큰 두 개의 객체를 선택하고, 그중에서 중심점이 중앙에 가까운 객체를 선택한다. 최종적으로, 면적이 가장 크고 중앙에 있는 객체가 중심점을 포함하게 되며, 그 중심점이 게이지 원의 중심점이 된다. 원의 중심좌표를 찾는 과정에 대한 전반적인 알고리즘을 그림 3에 제시하였다.

```

Algorithm 1 : Detection the Centroid of Circle
function finder_center(bwimage)
Input: binary image including centroid(bwimage)
Output: centroid(x, y)
1 MainObject = find the biggest object & fill holes in bwimage
2 Area = area(MainObject)
3 while (Area > (1/49)*size(bwimage))
4   open & erode in MainObject
5   number = re-count objects in MainObject
6   if number == 1
7     Area = re-area(MainObject)
8   else
9     choose the two biggest objects
10    MainObject = select the object close to the center point
11    Area = re-area(MainObject)
12  end
13 end
14 [x, y] = calculate the centroid of MainObject

```

Fig. 3. Algorithm for detecting the centroid of circle

### 3.3 Transforming rectangle into circle

아날로그 게이지 바늘의 위치를 측정하기 위해서, 원의 이미지를 극좌표를 사용하여 사각형 이미지로 변환해야 한다. 극좌표는 위에서 구한 원의 중심과 반지름을 이용해서 구한다. 반지름의 길이는 좌표의 범위, 이미지 길이에서 좌표의 범위의 차 중 가장 작은 값을 사용한다. 변환된 사각형은 세로길이가 반지름에 해당하고, 가로길이는 360픽셀의 크기를 가진다. 사각형의 세로 부분은 0부터 반지름의 크기를 반지름 구간으로 나누고 가로 부분은 0부터 360개의 구간으로 나눈다. 원주각이 1도씩 변환에 따라서 반지름의 크기만큼 x좌표와 y좌표에 해당하는 화소값을 읽어 와서 사각형 이미지에 저장한다. 그림 4는 직교좌표에 있는 한 점에 대해서 극좌표로 변환할 때의 과정을 나타낸다.

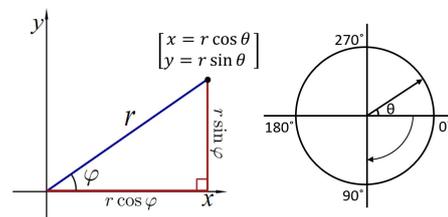


Fig. 4. Polar coordinates

### 3.4 Row region detection with scale

사각형 형태로 변환된 이미지를 살펴보면 아날로그 게이지에 표시된 눈금은 일정하고 규칙적인 형태를 나타내고 있다. 이 이미지에서 반복적인 특성과 주기성이 있는 영역을 찾기 위하여 본 논문은 다음 식(1)을 기반으로 하는 DMF(Distance Matching Function)를 사용한다[11].

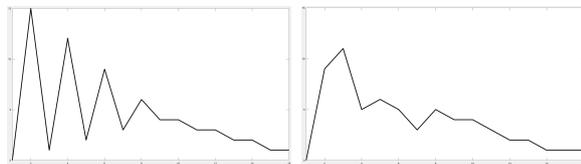
$$\lambda(\delta) = \sum_{i=1}^{N-\delta} [g(i) - g(i+\delta)]^2 \quad \text{or} \quad (1)$$

$$= \sum_{i=1}^{N-\delta} |g(i) - g(i+\delta)|, \quad N = \text{행의 픽셀 수}$$

식(1)에서 N은 사각형 이미지에 있는 하나의 행의 크기를 나타낸다. 즉, N = 360이다. 그리고 δ는 하나의 행 벡터에서 δ 픽셀만큼 오른쪽으로 이동한 결과를 나타낸다. DMF는 이미지의 각 행에 대하여 원본 벡터와 δ 픽셀만큼 이동된 벡터의 거리를 표시하는 함수가 된다.

DMF는 반복되는 패턴이 일정할 경우 정확한 주기를 나타낸다. 그림 5(a)는 16비트 이진 영상에서 눈금 영역의 DMF로 일정한 주기성을 나타내고 있다. 눈금이 있는 영역에서는 일정한 간격만큼의 주기를 확인할 수 있고, 반복되지 않는 구간에서는 주기가 나타나지 않는 것을 볼 수 있다.

반면에 그림 5(b)와 같은 불규칙한 반복 패턴에서는 주기성이 현저히 떨어지는 것을 확인할 수 있다. 실제 DMF의 응용에서는 반복 주기가 정확히 나타나지는 않는다. 따라서, 사각형 형태로 변환된 이미지에서 반복 패턴이 불규칙적으로 발생하고 이에 대한 주기성도 근사적으로 구해야 한다. 다음 알고리즘은 본 논문이 제안하는 불규칙한 반복 패턴의 주기성 추출 알고리즘이다.



(a) Periodic (b) Non-periodic  
Fig. 5. Example of periodicity and non-periodicity

눈금이 포함된 행 영역을 찾기 위해서 이진화 이미지의 행 벡터를 기준으로 각 행의 DMF를 계산하고 정규화한다. 계산된 DMF 값을 기준으로 행마다 극소점과 인덱스 그리고 극대점과 인덱스를 찾는다. 극대점의 판단 기준은 극소점의 가장 오른쪽에 자리 잡는 것으로 정한다. 극대점과 극소점의 차이를 구해서 극솟값의 위치와 진폭을 가지고 주기와 빈도수를 계산한다. 계산된 주기와 빈도수에 따라 내림차순으로 정렬하여 주기가 가장 큰 5개의 행을 눈금이 있는 가로 영역으로 추출한다. 그림 6은 눈금이 포함된 행 영역을 찾는 방법에 대한 알고리즘이다.

```

Algorithm 2 : Finding the Rows with Repeated Periodicity
function scale_row(rtimage)
Input: rectangular binary image(rtimage)
Output: 5 rows with repeated maximum periodicity
1 Dmf = calculate DMF on each row of the rectangular image
2 and normalize it
3 LocalMinimum = find local minimum of each Dmf
4 LocalMaximum = find local maximum of each Dmf
5 Period = calculate periodicity using right hand LocalMaximum
6 based on LocalMinimum
7 Save Period of each row(Dmf) and count it
8
9 Find the 5 rows with repeated maximum periodicity
    
```

Fig. 6. Algorithm for finding scaled rows

### 3.5 Column region detection with scale

눈금이 있는 열은 게이지 바늘의 위치를 기준으로 좌우로 탐색하며 눈금의 영역을 결정한다. 바늘의 위치를 계산하기 위해 이진화 사각형 이미지에서 모든 열의 히스토그램을 구한다. 히스토그램에서 나타나는 가장 큰 값이 위치한 열에 바늘이 있다는 것으로 결정한다. 3.4절에서 눈금 행으로 선택된 5개의 행을 OR 연산을 사용하여 하나의 최적화된 눈금 행으로 통합한다.

5개의 행에 대한 주요 눈금 주기는 OR 연산을 통해 하나로 통합된 행을 대상으로 눈금 범위를 측정한다. 바늘이 위치한 열 영역을 기준으로 일정 주기만큼 좌우로 이동하며 탐색한다. 탐색할 때 필요한 일정 주기는 3.4절에서 찾은 주기를 비교하여 그중에서 가장 많은 빈도를 차지하는 주기로 정한다.

먼저, 바늘의 오른쪽부터 눈금 범위를 탐색을 진행한다. 주기 크기에 해당하는 픽셀값에 눈금에 해당하는 흰색이 나타나지 않고 전부 검은색으로 되어있다면, 이것은 눈금의 범위를 벗어난 것으로 볼 수 있다. 눈금의 범위를 벗어나기 전에 확인했던 인덱스가 바늘의 오른쪽에 해당하는 눈금의 최대 범위 인덱스이다. 바늘의 왼쪽을 탐색할 때도 이와 같은 방법으로 진행한다. 그림 7은 위에서 구한 눈금의 주기를 이용하여 바늘의 위치를 기준으로 한 눈금의 좌우 최대 범위를 탐색하는 알고리즘이다.

```

Algorithm 3 : Finding the Scaled Columns
function scale_column(scaledimage)
Input: binary row image by OR operation to the 5 selected rows
Output: [left_max, right_max] in scaled columns
1 Period = select the most frequent period of the 5 selected rows
2 // find right_max
3 right_image = 2*scaledimage
4 right_max = hand_position
5 while right_max <= (hand_position + 360)
6     searching = right_image(right_max+1: right_max+1.5*period)
7     last_index = find the last index in searching
8     if no last_index |
9         exit
10    else
11        right_max = right_max + last_index
12    end
13    if right_max >= (hand_position + 360)
14        right_max = hand_position + 360
15    end
16 // find left_max
17 // It is similar to above code. But its direction is opposite.
    
```

Fig. 7. Algorithm for finding scaled columns

그러나 아날로그 게이지의 눈금은 원형으로 표시되기 때문에 단순히 바늘을 중심으로 눈금 범위를 좌우로 탐색하게 된다. 그림 8에서 바늘의 왼쪽을 탐색했을 때, 최소 눈금 값을 찾을 수 있지만, 반대로 바늘의 오른쪽을 탐색했을 때 실제 최대 눈금 값이 아닌 가장 오른쪽에 있는 값을 인식하여 눈금의 범위를 정확하게 구할 수 없는 문제가 발생한다.

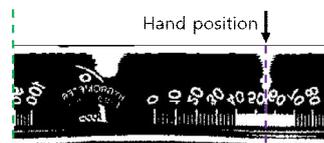


Fig. 8. Scale range search

이러한 문제를 해결하기 위해 그림 9와 같이 전체 이미지를 복사한 후에 탐색 진행 방향에 맞게 붙여넣어 이미지를 확장한다. 예를 들면, 오른쪽으로 탐색을 진행했을 때 왼쪽에 있는 그림이 오른쪽으로 이어 붙여지면서 기준에 왼쪽에 자리 잡고 있어서 탐색할 수 없었던 최대 눈금 값을 찾을 수 있게 된다.

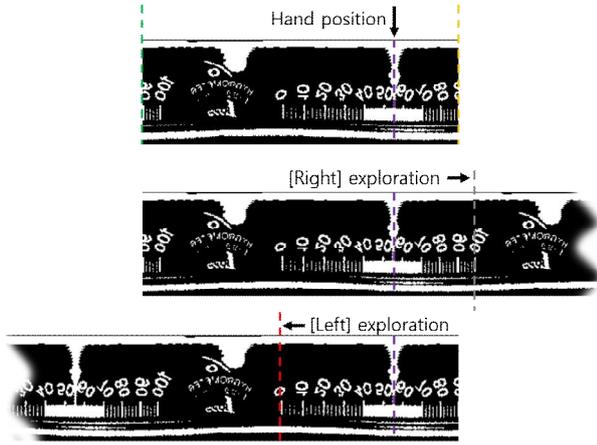


Fig. 9. Process of scale range search

### 3.6 Calculating analog gauge value

아날로그 게이지의 최솟값과 최댓값은 기기에 따라 다르므로 사용자가 직접 최솟값과 최댓값을 지정한다. 그리고 눈금이 게이지 안에서 전 영역(360도)에 걸쳐 있는지 그렇지 않은지를 판단하고 눈금의 시작 값이 음수인지 양수인지 확인하여 게이지 값을 다르게 계산한다.

눈금이 전 구간에 분포한 경우는 대부분 12시 방향에 눈금의 시작과 끝이 있다. 본 논문에서 이미지를 탐색할 때는 12시 방향을 기준으로 3시 방향부터 1도씩 회전하며 이미지 탐색을 진행한다. 바늘이 가리키는 눈금은 바늘의 위치에 90도를 더하여 눈금의 최솟값과 최댓값에 맞추어 값을 계산하여 출력한다.

눈금이 일부분에 분포해 있는 경우는 그림 10과 같이 사용자가 입력한 아날로그 게이지의 최솟값과 최댓값, 그리고 바늘이 가리키는 값에서 최솟값과 최댓값까지의 거리를 비율로 계산하여 게이지의 값을 계산하여 출력한다. 최솟값이 음수일 경우 최솟값을 절댓값으로 바꾸어서 현재 가리키고 있는 눈금 값에서 차를 구하여 값을 계산한다. 그림 11은 눈금 값을 계산하는 과정에 대한 알고리즘을 나타낸 것이다.

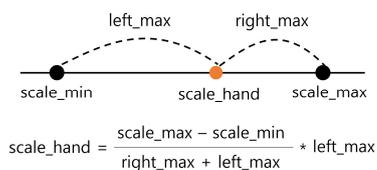


Fig. 10. Calculate scale hand value

Algorithm 4 : Calculate Hand Value	
function	calculate_hand(left_max, right_max, hand_position)
Input:	left_max, right_max, hand_position in previous steps scale_max, scale_min inputed by users
Output:	scale_hand evaluated by hand_position
1	if mod(abs(left_max-right_max), 360) < period
2	// 360 circled scale
3	scale_hand = ((hand_position+90)/360)*scale_max;
4	else // partitioned scale
5	scale_hand = ((scale_max*left_max) + (scale_min*right_max))/(left_max + right_max);
6	if scale_min > 0 % the scale starts in negative number
7	scale_hand = scale_min + scale_hand;
8	else % the scale starts in positive number
9	scale_hand = scale_hand - abs(scale_min);
10	end
11	end
12	end
13	end

Fig. 11. Algorithm for calculating hand value

## IV. Experiments

아날로그 게이지의 형태는 전부 유사하게 보여도 중심점의 위치나 눈금의 크기, 눈금의 범위 등 각각 다르므로 본 논문에서는 원형 아날로그 게이지 다양한 종류의 이미지를 가지고 이미지의 중심점을 인식하는 것과 눈금의 시작과 끝의 범위 인식 및 바늘이 가리키는 게이지값을 확인하는 테스트를 진행하였다.

### 4.1 Each step of proposed algorithm

이미지 전처리 단계에서는 이미지를 250x250의 해상도로 변환하고 RGB 영상을 그레이스케일로 변환 후에 그레이스케일 영상에서 전체적인 픽셀값들의 명도 레벨에 맞추어서 이진화를 시행한다. 이진화는 Otsu's 방법을 적용한 것이다. 그림 12는 RGB 영상에서 이진화 영상으로의 진행과정을 보여주고 있다.

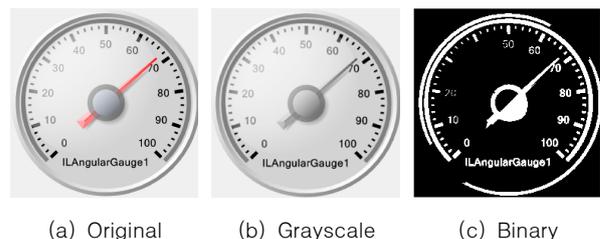


Fig. 12. Image preprocessing

이진화 이미지를 3x3 이미지로 나누었을 때 그림 13을 보면 게이지의 중심부가 3x3의 중앙에 포함된다. 3x3의 중앙 이미지를 가져와서 객체의 중심을 찾을 때 형태학적 연산인 침식과 열림 연산을 통해서 면적을 축소하면서 더욱더 정확한 중심좌표를 구하게 된다. 그림 14는 이와 같은 과정을 통해서 이미지의 중심부가 축소되는 과정을 보여준다.

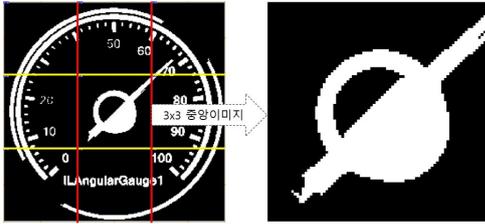


Fig. 13. Cropping center image

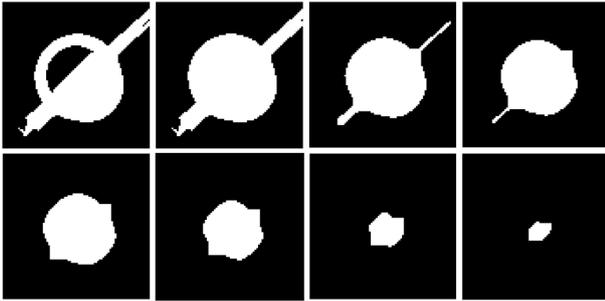
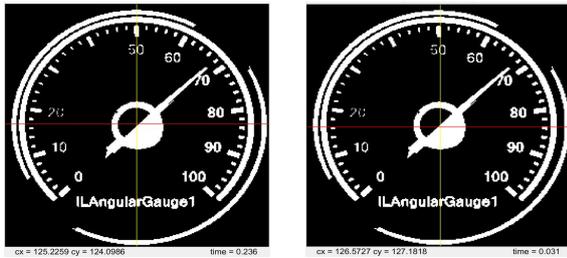


Fig. 14. Image reduction using morphological operation

일반적으로 원의 중심좌표를 구할 때 사용하는 방법으로 원형검출 허프변환이 있다. 그림 15는 원형검출 허프변환과 본문에서 사용한 방법을 비교한 것이다. 중심좌표는 둘 다 비슷하게 구한 것 같지만, 좌표를 구하는 계산 시간을 비교해보면 원형검출 허프변환 처리시간과 형태학적 변환 처리시간이 서로 차이가 나는 것을 확인할 수 있다.



(a) Circular hough transform (b) Morphological operation

Fig. 15. Finding center of circle

DMF를 이용하여 눈금 범위를 구하기 위해서는 원형의 형태를 직사각형으로 변환하여 주기성을 확인하는 작업이 필요하다. 극좌표를 활용하여 3시 방향부터 시계방향으로 1도씩 회전하면서 직사각형의 이미지로 변환한다. 변환된 이미지의 행에 대해 주기성을 분석해서 가장 많은 주기성을 띄는 5개의 행에 대해 눈금이 위치한 행으로 검출한다. 그림 16은 그림 15의 원형 형태의 이미지를 극좌표를 활용하여 사각형으로 변환된 결과이며, 빨간 선으로 된 부분은 주기성을 분석해서 눈금이 포함되어 있을 5개의 행에 대해 표시한 것이다.

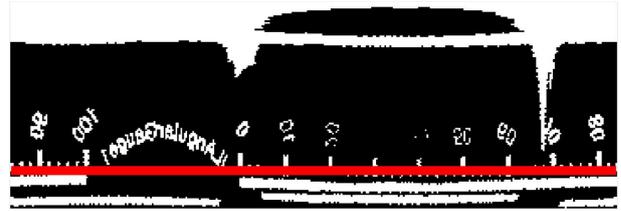


Fig. 16. Converting circle to rectangle using polar coordinate

눈금 행으로 검출된 5개의 행에 대해 OR 연산을 진행하고 바늘의 위치를 찾은 다음, 위에서 구한 눈금의 주기만큼 좌측으로 먼저 탐색을 진행한다. 그림 17처럼 눈금이 이어져 있지 않고 떨어져 있는 경우는 이미지를 좌우로 복사해서 탐색을 진행한다. 주기만큼의 영역에서 픽셀값이 나타나지 않을 때까지 진행하며, 픽셀값이 나타나지 않는 부분의 인덱스를 바늘을 기준으로 눈금의 좌측 최대 범위로 지정한다. 이 과정을 좌측 방법과 같이 우측으로 진행하여 바늘을 기준으로 눈금의 우측 최대 범위로 지정하여 최종적으로 눈금의 전체 범위를 지정한다.

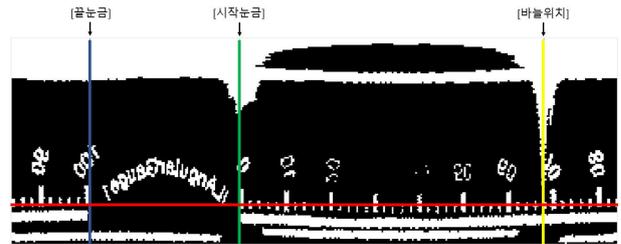


Fig. 17. Detection of scale range and hand

시작 눈금과 끝 눈금 그리고 바늘 위치의 인덱스를 통해서 실제 눈금의 최솟값과 최댓값을 이용한다. 그런 다음, 현재 게이지의 바늘이 가리키고 있는 값을 눈금이 전 범위에 걸쳐져 있는 경우와 그렇지 않은 경우를 구분하여 게이지 값을 계산한다. 그림 18은 눈금이 전 범위에 걸쳐져 있는 경우와 그렇지 않은 경우에 대해서 나타내고 있다.



(a) Partial range scale (b) Full range scale

Fig. 18. Example of analog gauge recognition

#### 4.2 Test of various analog gauges

본 논문은 14개의 다양한 원형 아날로그 게이지 영상을 대상으로 실험하였다. 표 1은 원형검출 허프변환을 사용하여 중

Table. 1. Result of finding center (unit : sec)

Img No.	1	2	3	4	5	6	7
Circular Hough Transform	0.195	0.291	0.195	0.291	0.173	0.178	NaN
Morphological Operation	0.029	0.036	0.028	0.028	0.034	0.017	0.029
Img No.	8	9	10	11	12	13	14
Circular Hough Transform	0.160	0.186	0.149	NaN	0.196	0.145	0.184
Morphological Operation	0.023	0.019	0.006	0.012	0.017	0.012	0.012

심좌표를 찾을 때와 형태학적 변환을 통해 중심좌표를 찾는 두 방법에 대해 비교한 결과를 나타내고 있다.

원형검출 허프변환은 평균 0.195s 시간이 소요되었고, 형태학적 변환은 평균 0.022s 시간이 소요되었다. 원형검출 허프변환은 원의 방정식에 들어가는 수를 모두 계산해야 해서 시간이 많이 소요되며 정확한 원의 형태가 아닐 경우 원형검출 허프변환을 통해 중심점을 인식하지 못하는 경우도 발생하였다.

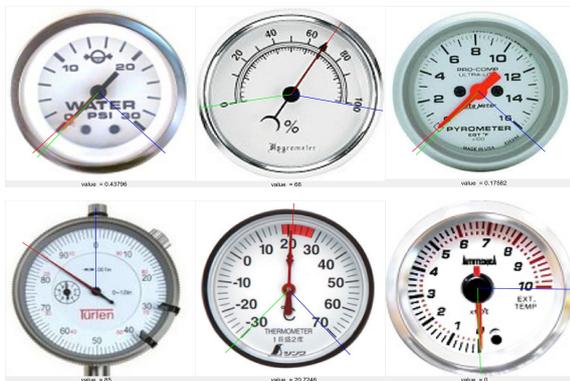
그림 19는 여러 이미지에 대해서 중심좌표와 눈금 범위를 인식해서 게이지 값을 출력한 결과물이다. (a)는 눈금의 시작과 끝 그리고 바늘을 잘 인식해서 게이지 값을 정확하게 출력하는 경우이다. (b)는 바늘은 잘 인식하지만, 눈금 바로 옆에 있는 상표나 노이즈에 의해서 눈금 범위를 벗어나서 인식하는 경우이다. 그러나 눈금의 범위에 대한 한쪽 범위는 비교적 정확하게 인식하고 다른 한쪽은 부정확하게 인식하여 근사적인 게이지 측정값을 얻을 수 있었다. (c)는 바늘과 눈금 범위 모두 인식하지 못하고 있다.



(c) Bad recognition

Fig. 19. Experiment result of gauge recognition

전체적으로 이미지가 어둡거나 전처리 과정에서 노이즈가 발생하게 되면, 이미지를 이진화했을 경우 눈금이 안 보이는 경우 혹은 눈금과 배경이 겹치게 되어 눈금 인식에 어려움이 발생하며, 또 다른 이유로는 원형이 아니고 타원형이면 극좌표를 이용해서 원형의 이미지를 직사각형 형태로 변환할 때 그림 20과 같이 눈금의 형태가 반듯하지 않아 눈금과 눈금 사이의 주기성을 분석할 때 문제가 발생하게 된다.



(a) Good recognition



(b) Approximate recognition



Fig. 20. Result for not finding center of circle

## IV. Conclusions

본 논문에서는 형태학적 연산을 사용하여 원을 인식하는 방법, 주기를 이용하여 눈금의 위치와 범위를 구하는 방법에 대해 소개하고, 원형검출 허프변환과 형태학적 연산을 사용했을 때의 차이를 비교하고, 여러 종류의 원형 아날로그 게이지를 대상으로 바늘이 가리키는 게이지 값을 출력하는 과정에 대해서 실험을 진행하였다. 원을 직사각형 형태로 변환시키기 위해서는 원의 중심좌표가 필요한데 기존의 연구들은 원형검출 허프변환을 사용했지만, 정확한 원의 형태가 아닐 경우 중심좌표를 구하지 못하는 경우도 발생했다. 본 논문은 형태학적 연산을 적용하여 중심부를 인식하여 중심 객체를 축소하는 방법을 이용하여 중심좌표를 계산했으며, 원형검출 허프변환을 사용할 때보다 계산 시간도 현저히 줄어들고 더욱더 정확한 중심좌표를 구할 수 있게 되었다.

또한, 주기성을 활용하여 게이지를 가리키는 바늘의 위치와 눈금의 범위를 구하기 때문에 따로 시작 눈금과 끝 눈금의 위치를 파악하지 않아도 되는 장점이 있다. 눈금의 범위를 탐색할 때에도 눈금이 한쪽에 치우쳐있거나 좌우로 나누어져 있더라도 탐색하는 방향에 따라서 이미지를 복사해서 좌우로 탐색하기 때문에 눈금의 범위를 정확하게 인식할 수 있었다. 그리고 눈금이 360도에 걸쳐서 분포한 경우와 그렇지 않은 모든 경우에 대해 알고리즘을 구성해서 다양한 원형 아날로그 게이지를 인식할 수 있었다.

연구 결과는 향후 원형 아날로그 게이지뿐만 아니라 다양한 형태의 아날로그 게이지 인식에도 적용할 수 있어 응용성이 우수하며, 눈금의 범위를 찾기 위해 사용된 DMF를 가지고 반복되는 무늬나 패턴을 인식하는 연구에도 많이 활용될 것으로 생각한다.

## REFERENCES

- [1] Kim. J. W, "Current Status and Development Plan for Smart Factory Education," The Korea Contents Association Review, Vol. 15, No. 2, pp. 25-29, June 2017.
- [2] Internet of Things(IoT), [https://en.wikipedia.org/wiki/Internet\\_of\\_things](https://en.wikipedia.org/wiki/Internet_of_things)
- [3] Types of Gauges, <https://www.thomasnet.com/articles/pumps-valves-accessories/types-of-gauges>
- [4] Park. B. S, "Design and Implementation of M2M-based Smart Factory Management Systems that controls with Smart Phone," Journal of the Korea Society of Computer and Information, Vol. 16, No. 4, pp. 189-196, Apr. 2011.
- [5] S. J. K. Pedersen, "Circular Hough Transform," Aalborg University, Vision, Graphics and Interactive Systems, Nov. 2007.
- [6] M. K. Gellaboina, G. Swaminathan and V. Venkoparao, "Analog dial gauge reader for handheld devices," IEEE Industrial Electronics and Applications(ICIEA), pp. 1147-1150, 2013.
- [7] D. L. de Lima, G. A. S. Pereira, and F. H. de Vasconcelos, "A computer vision system to read meter displays," 16th IMEKO TC4 Symposium Exploring New Frontiers of Instrumentation and Methods for Electrical and Electronic Measurements, 2008.
- [8] E. C. Alegria and A. C. Serra, "Automatic calibration of analog and digital measuring instruments using computer vision," IEEE Transactions on Instrumentation and Measurement, pp. 94-99, 2000.
- [9] X. Ye, D. Xie and S. Tao, "Automatic Value Identification of Pointer-Type Pressure Gauge Based on Machine Vision," JCP, pp. 1309-1314, 2013.
- [10] V. Asha, N. U. Bhajantri and P. Nagabhusan, "Periodicity Extraction using Superposition of Distance Matching Function and One-dimensional Haar Wavelet Transform," arXiv preprint, arXiv:13 11.3808, 2013.
- [11] Oh. G. H, Lee. S. Y and Shin. S. Y, "Fast determination of textural periodicity using distance matching function," Pattern Recognition Letters 20.2, pp. 191-197, 1999.

## Authors



Jin-kyu Ryu received the B.S. degrees in information technology and engineering from the Chonbuk National University, Republic of Korea, in 2016, M.S. degrees in computer engineering from the Chonbuk National University, Republic of Korea, in

2018, respectively. His research interests include image processing and pattern recognition.



Young-Tae Kwak received the B.S., M.S., and Ph.D. degrees in computer engineering from the Chungnam National University, Republic of Korea, in 1993, 1995, and 2001, respectively. He joined the faculty of the Chonbuk National University in 2002. His

research interests include pattern recognition and neural networks.