대한조선학회 특별논문집
2009 년 9 월

Special Issue of the Society of Naval Architects of Korea
pp. 115-122, September 2009

# 이동식 자동 마킹 시스템을 위한 문자열 분할 알고리즘

박두경† *, 한상동*, 김호구*, 안정기*, 김대경*

삼성중공업㈜ 산업기술연구소 자동화연구센터

# Character Split Algorithm for an Automated Marking System with a Moving Ink-head

Doo-Kyung Park † *, Sang-Dong Han *, Ho-Gu Kim *, Jung-Ki* Ahn and Dae-Kyung Kim *

Institute of Industrial Technology/Automation System Research, Samsung Heavy Industries*

Abstract

In shipbuilding process, automated marking systems are used to print steel part information on the steel plate. Especially, an automated marking system with a moving ink-head prints characters on each block, which divide the steel plate into a fixed size. Therefore it is necessary to split characters, because some characters can be extended into several blocks. Particularly the algorithm, which splits characters, must guarantee its printing quality in any direction since characters can be printed by any angle.

※Keywords: Automated marking system(자동 마킹 시스템), Characters split(문자열 분할)

## 1. 서론

하나의 선박을 제조하는 데에는 많은 양의 강판(Steel Plate)이 필요하다. 강판에는 강재명, 부재명과 같이 전처리 이후 공정에 필요한 정보를 문자열 형식으로 마킹함으로써 강판을 효율적으로 관리할 수 있다. 특히, 강판에 정보를 담고 있는 문자를 자동으로 마킹할 수 있으면 마킹 속도가 증가되고 작업 공수를 줄일 수 있어 생산성을 향상시킬 수 있다. 그러므로 최근에는 자동 마킹 시스템의 도입 및 활용이 보편화 되어가고 있다.

†교신저자: dookyung.park@samsung.com, 055-630-4296

자동 마킹 시스템은 크게 고정식 자동 마킹 시스템과 이동식 자동 마킹 시스템으로 분류된다.

먼저 고정식 자동 마킹 시스템에 대해 설명하면, 전처리부에 의해 이송되는 강판의 종류를 감지하는 강재 센서 및 상기 강판에 문자를 인쇄하여 마킹하는 잉크 헤드를 포함한다. 잉크 헤드는 거더에 고정되어 있으며, 제어부는 잉크 헤드에 구비된 잉크 노즐 중에서 마킹하고자 하는 문자의 위치에 대응하는 잉크 노즐의 on/off 를 제어하여 잉크를 강판에 분사하게 된다. 그러나, 고정식 자동 마킹 시스템은 고가의 잉크 헤드가 강판의 폭 방향 전 영역에 설치되어야 하므로 많은 비용이 소요된다. 또한 잉크 헤드에는 많은

수의 잉크 노즐이 구비되기 때문에, 상기 잉크 노즐이 자주 막히게 된다.

이와 같은 단점을 보완하고자 이동식 잉크 헤드 타입의 자동 마킹 시스템(이하 ‘ 이동식 자동 마킹 시스템)가 등장하였다. 이동식 자동 마킹 시스템은 강판의 폭보다 좁은 폭을 가지는 잉크 헤드를 좌우 방향으로 이동시켜 문자를 마킹하게 된다. 이와 같은 이동식 자동 마킹 시스템은 고가의 잉크 헤드의 크기를 줄일 수 있어서 장치 비용을 절감할 수 있으며, 적은 수의 잉크 노즐을 가지므로 잉크 노즐이 자주 막히는 문제점을 해결할 수 있다 Ahn(2007).

그러나 이동식 자동 마킹 시스템은 강재를 고정된 크기를 갖는 블록으로 나눈 후 잉크 헤드를 각 블록 단위로 이동시켜 문자를 마킹하게 되는데, 만약 강판의 정보를 나타내는 문자열이 하나의 블록 안에 들어가지 못하는 경우 문자열을 나누어서 인쇄해야 한다. 이와 같이 강판의 정보를 나타내는 문자열을 나누어서 인쇄할 경우 문자열에 단차가 발생하여 사용자가 문자열을 파악하지 못하거나 문자열을 파악하는데 있어 많은 시간이 걸리는 문제점이 있다.

따라서 본 논문은 먼저 이동식 잉크 헤드가 적용된 자동 마킹 시스템에 대해 설명하고 그 다음으로는 문자열이 여러 개의 블록에 걸쳐 있을 때 문자열을 분할하는 방법에 대해 이야기할 것이다. 그리고 마지막으로, 실제 산업 현장에서 적용된 사례에 대해 소개하고 결론을 맺을 것이다.

## 2. 이동식 자동 마킹 시스템

Fig. 1 은 이동식 자동 마킹 시스템의 전체적인 구성도다. 이동식 자동 마킹 시스템은 강판을 자동으로 이송시키는 컨베이어 롤러(1), 컨베이어 롤러에 의해 이송되는 강판의 반입상태와 형태를 검출하는 강재 센서(2), 이동하는 강판에 강판의 정보를 나타내는 문자열을 마킹하는 마킹장치부(3), 각종 제어장치를 한군데로 모아 놓은 제어부(4), 강판에 마킹 될 이미지를 생성하는 운영부(5), 전처리 공정으로 투입되는 강판 종류를 결정하는 공정 관리부(6)로 구성된다.

제어부는 텍스트 정보를 그래픽 정보로 변환하는 마킹 정보 변환 프로그램, 마킹 위치 변화에 따른 그래픽 제어 시스템, 마킹 장치부 제어프로그램, 그리고 각종 센서를 제어하는 센서 제어프로그램 등을 포함하여 구성된다. 또한, 마킹 정보 변환 프로그램 및 마킹 장치부 제어 프로그램의 정보를 바탕으로 각 축 모터를 제어함으로써 이동식 잉크 헤드의 동작을 제어하게 된다.

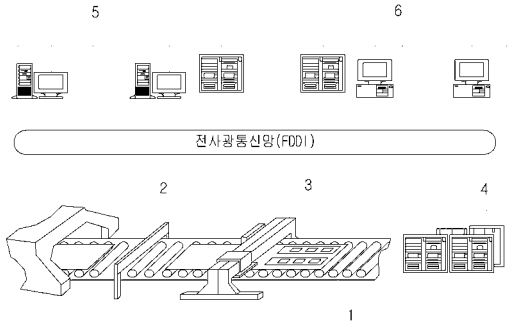


Fig. 1 Structure of the automated marking system with a moving ink-head.

이동식 자동 마킹 시스템의 동작에 대해 설명하면 다음과 같다.

먼저 설계에서 강판에 마킹할 부재 정보(steel part information)를 운영부에서 마킹을 하기 전에 입력한다. 그리고 공정관리부에 의해 마킹될 강판이 컨베에 롤러로 투입되면, 강재 정보(steel plate information)가 운영부로 전달된다. 이 후 강재 센서가 강판을 인식하고 강재 진입 위치 및 각도 등의 강재 정보를 측정한 후 해당 계측 데이터(measured data)를 운영부로 보내준다. Fig. 2 와 같이 운영부는 계측 데이터를 이용 출력 문자열의 위치 및 각도를 보정하여 출력을 위한 이미지 데이터(image data file)를 생성한 후 잉크 헤드로 전송한다. 최종적으로 마킹 장치부에서는 제어부의 제어에 의해 항상 X 축 방향으로 왕복하면서 강판에 마킹을 실행한다 Ryu et al.(2008).

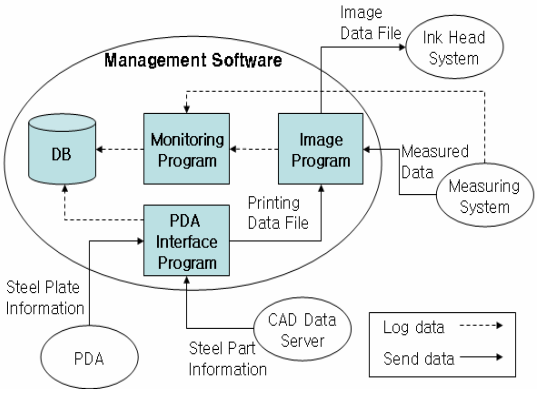


Fig. 2 Data flow of management system

이 때, 마킹 장치부의 잉크 헤드가 좌측 또는 우측 방향 중 일 방향으로 이동할 때 마킹 가능한 영역을 ‘ 블록’이라고 정의한다. 따라서, 잉크 헤드의 Y 축 방향의 길이가 987mm 라고 하면, 하나의 블록의 Y 축 방향 길이도 987mm 가 된다.

## 3. 문자열 분할 알고리즘

### 3.1 블록 설정

일반적으로 문자열이 하나의 블록에 전부 마킹할 수 있는 경우는 문제가 없으나, Fig. 3 에 표시된 바와 같이 문자열이 하나의 블록을 넘어 2 개 이상의 블록에 걸쳐 존재하는 경우, 제어부가

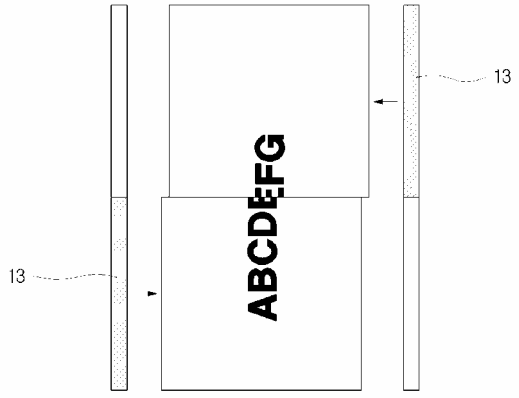


Fig. 3 A broken character ‘ E’

잉크 헤드를 제어할 때 발생하는 제어오차 및 이동하는 강판이 좌우 방향으로 움직임에 따라 발생하는 오차 등에 의해 블록 간의 경계부에 마킹되는 문자의 식별이 어렵다. Fig. 3 은 문자열을 구성하는 문자 중 ‘ E’가 나뉘어 출력되어 식별이 어려운 경우다.

이와 같은 문제점은 블록 중첩 영역을 설정함으로서 해결 할 수 있다.

여기서 ‘ 블록 중첩 영역’이란 잉크 헤드가 중첩하여 문자를 인쇄할 수 있는 영역을 말한다.

즉, 잉크 헤드는 블록 중첩 영역을 두 번 지나가면서 문자를 마킹하게 된다.

블록 중첩 영역의 범위는 문자 마킹 조건과 설비 환경에 따라 다르게 설정할 수 있다. 블록 중첩 영역의 범위를 너무 넓게 설정하게 되면 하나의 강판에 문자열을 마킹하기 위해 잉크 헤드가 좌우 왕복 운동을 해야 하는 횟수가 많아질 수 있으며, 중첩 영역을 너무 좁게 설정하게 되면 강판에 마킹해야 하는 문자열을 구성하는 문자의 크기를 작게 해야 하기 때문이다. 상기 블록 중첩 영역은 마킹할 최대 문자크기, 문자열 각도, 이동 강판의 기울기 변형량을 고려하여 결정하는 것이 바람직하다. 이와 같은 블록 영역 구분에 대한 사례가 Fig. 4 에 나타나 있다.

이번 사례에서는 하나의 블록의 Y 축 길이가 987mm 라고 할 때, 대략 37mm 정도를 블록 중첩 영역으로 설정했다. 잉크 헤드가 처음으로 좌우 왕복 운동을 하는 첫 번째 블록에서는 앞단에만 중첩 영역이 존재 하지만, 2 번째 이후의 블록에서는 양단에 중첩 영역이 존재하게 된다.

이 때, 첫 번째 블록 하단의 Y 좌표를 0 으로 설정하면 첫 번째 블록 상단의 Y 좌표는 잉크 헤드의 길이인 987 가 된다. 그리고, 두 번째 블록 하단의 Y 좌표는 첫 번째 블록 상단의 Y 좌표인 987 에서 블록 중첩 영역의 길이인 37 을 뺀 950 이 된다.

그리고, 두 번째 블록 상단의 Y 좌표는 두 번째 블록 하단의 좌표인 950 에서 잉크 헤드의 길이인 987 을 더하면 1937 이 된다.

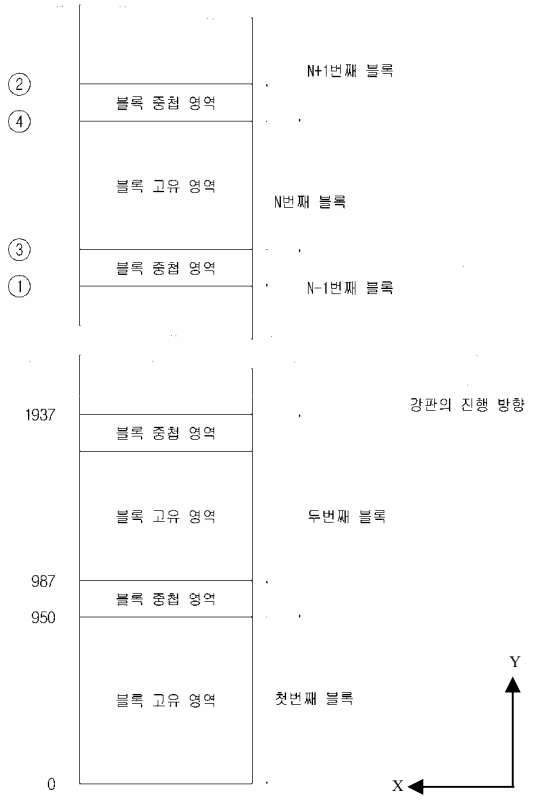


**Fig. 4** Individual area and overlap area of blocks

이와 같은 방식을 통해 N 번째 블록에서 블록 하단 및 상단의 Y 좌표를 계산할 수 있다.

Fig. 4 에서 N 번째 블록을 예로 들어 설명하면, ① 지점은 N 번째 블록의 최 하단이고, ② 지점은 N 번째 블록의 최 상단이고, ③ 지점은 N 번째 블록의 블록 고유 영역의 상단이고, ④ 지점은 N 번째 블록의 블록 고유 영역의 하단이 된다.

Fig. 5 는 첫 번째 블록과 N 번째 블록에 존재할 수 있는 문자열의 배치와 방향을 기준으로 문자열들을 그룹으로 나누어 분류한 것이다. Fig. 5 에서 화살표의 길이는 문자열의 길이를 나타내고, 화살표의 방향은 문자를 읽는 방향을 나타낸다.

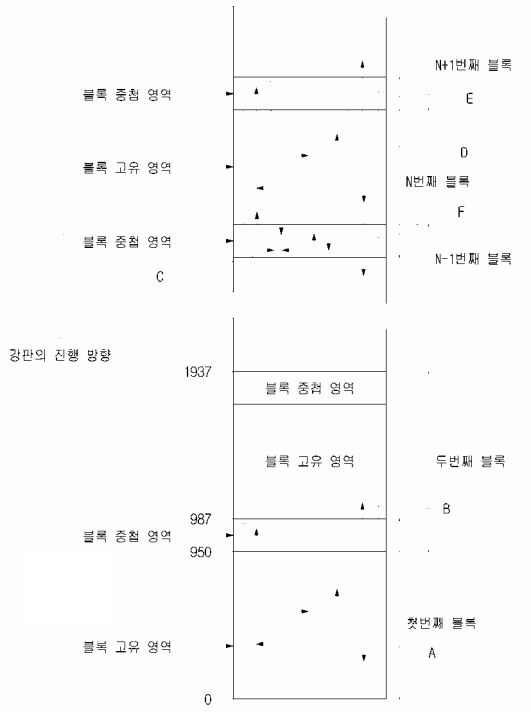


**Fig. 5** Grouping of characters

### 3.2 문자열 분류 및 배치

Fig. 6 는 문자열을 그룹별로 분류하는 방법을 나타낸다.

잉크 헤드가 문자열을 마킹해야 하는 블록(이하 ‘ 작업 블록’ )이 첫 번째 블록인지를 판단한다.

만일 작업 블록이 첫 번째 블록이라면 문자열이 모두 첫 번째 블록 내에 존재하는지를 판단한다. 이 때 문자열이 모두 첫 번째 블록 내에 존재하면 A 그룹으로 분류하고, 문자열이 모두 첫 번째 블록 내에 존재하지 않으면 B 그룹으로 분류한다. 문자열이 모두 이전 블록과의 블록 중첩 영역 내에 존재하면 C 그룹으로 분류한다.

문자열이 모두 이전 블록의 블록 중첩 영역 내에 존재하지 않으면 다시 문자열이 모두 N 번째 블록 내에 존재하는지를 판단한다. 이 때 문자열이 모두 N 번째 블록 내에 존재하게 되면 D 그룹으로 분류한다.

다음으로 문자열이 N 번째 블록에서 이전 또는 다음 블록으로 넘어가는 지를 판단한다. 이 때, 문자열이 다음 블록으로 넘어가면 E 그룹으로 분류하고, 문자열이 이전 블록으로 넘어가면 F 그룹으로 분류한다.

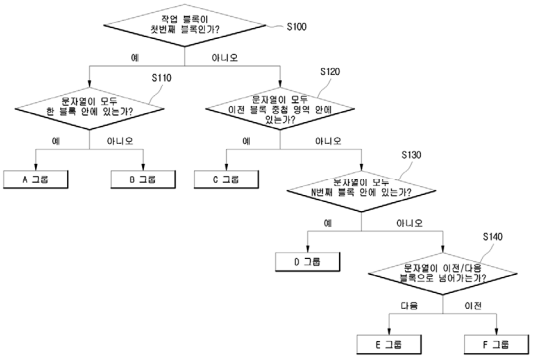


Fig. 6 A classification conditions of groups

위와 같은 방법으로 A 부터 F 그룹으로 분류된 문자열 분류 기준에 따라 문자열을 재배치한다. 재배치 방법은 아래와 같다.

A 그룹은 현재 블록(첫 번째 블록)의 미리 설정된 좌표에 문자열을 배치한다.

B 그룹은 문자열 중에서 첫 번째 블록의 최상단을 넘어가는 최초의 문자(이하 ‘ S 번째 문자’)를 검색 후, 문자열의 첫 문자부터 S 번째 문자 바로 이전 문자(S-1 번째 문자)까지는 첫 번째 블록에 미리 설정된 좌표에 배치한다. S 문자부터 문자열 끝까지는 S 문자를 마킹하는 좌표를 재계산한 후 두 번째 블록에 배치한다.

C 그룹은 이전 블록(N-1 번째 블록)의 미리 설정된 좌표에 문자열을 배치한다.

D 그룹은 현재 블록(N 번째 블록)의 미리 설정된 좌표에 문자열을 배치한다.

E 그룹은 문자열 중에서 N 번째 블록의 최상단을 넘어가는 최초의 문자(S)를 검색 후, 문자열의 첫 문자부터 S 문자 바로 이전 문자(S-1)까지는 N 번째 블록에 미리 설정된 좌표에 배치한다. S 문자부터 문자열 끝까지는 S 문자를 마킹하는 좌표를 재계산한 후 다음 블록(N+1 번째 블록)에 배치한다.

F 그룹은 문자열 중에서 N 번째 블록의 최상단을 넘어가는 최초의 문자(S)를 검색 후, 문자열의 첫 문자부터 S 문자 바로 이전 문자(S-1)까지는 N 번째 블록에 미리 설정된 좌표에 배치한다. S 문자부터 문자열 끝까지는 S 문자를 마킹하는 좌표를 재계산한 후 다음 블록(N+1 번째 블록)에 배치한다.

### 3.3 문자열 좌표 재 계산

N 번째 블록에서 다음 블록(N+1 번째 블록) 또는 이전 블록(N-1 번째 블록)으로 문자열의 일부가 넘어갈 때 넘어가는 문자를 N 번째 블록에 마킹되는 문자열과 분할하여 다음 블록(N+1 번째 블록) 또는 이전 블록(N-1 번째 블록)에 마킹하게 되는데, 분할되어 마킹될 문자열의 시작 좌표를 구하는 방법이 Fig. 7 에 나타나 있다.

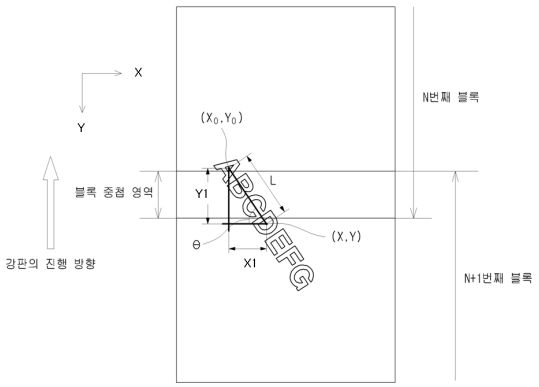


Fig. 7 Calculation of the location for split characters ‘ DEFG’

Fig. 7 에 도시된 바와 같이 문자열이 마킹되어야 할 시작 좌표를 ($X_0$, $Y_0$)라고 할 때, 문자열을 구성하는 문자 중 D 가 N 번째 블록의 블록 중첩 영역과 N+1 번째 블록의 블록 고유 영역의 경계부에 위치하고 있다.

따라서, 상기 문자 D 및 D 이후의 문자인 EFG 는 N+1 번째 블록에서 분할하여 마킹하게

된다.

이 때, 문자열 DEFG 를 분할하여 마킹할 때, 분할하여 마킹하는 문자열의 최초 문자인 D 의 좌표를 구하는 공식은 아래와 같다.

$$X = X_0 + \cos\theta \cdot \text{문자크기} \cdot \text{문자수}$$
$$Y = Y_0 + \sin\theta \cdot \text{문자크기} \cdot \text{문자수}$$

$(X, Y)$: 분할된 문자열의 시작좌표
$(X_0, Y_0)$: 문자열의 시작좌표
$\theta$: Y축과 문자열이 형성하는 축이 이루는 각도
문자크기: 하나의 문자와 이웃하는 문자의 중심간 거리
문자수: 문자열의 첫번째 문자부터 다음 블록으로 넘어가는 문자의 이전 문자까지의 문자수

상기 식에서 문자 크기와 문자 수를 곱한 값은 문자열의 최초 문자인 A 의 중심과 분할되어 마킹되어야 할 최초 문자인 D 의 중심 간의 거리(L)가 된다. 이 거리(L)에 $\cos\theta$ 를 곱한 값은 X 축 좌표의 증분인 $X_1$, $Y_1$ 이 되고, 이 거리에 $\sin\theta$ 를 곱한 값은 Y 축 좌표의 증분인 $Y_1$ 이 된다. 따라서, 문자열의 시작 좌표($X_0$, $Y_0$)에서 이와 같은 증분($X_1$, $Y_1$)을 더함으로써 분할된 문자열의 시작좌표(X, Y)를 구할 수 있고, 상기 분할된 문자열의 시작좌표(X, Y)부터 분할된 문자열을 마킹함으로서 N 번째 블록의 블록 중첩 영역과 N+1 번째 블록의 블록 고유 영역의 경계부에 마킹되는 문자의 인쇄품질이 저하되는 것을 방지할 수 있다. 따라서, 사용자가 강판에 마킹된 문자열을 식별하지 못하는 문제점을 해결할 수 있게 된다.

## 4. 적용 사례

앞서 설명한 문자열 분할 알고리즘을 실제 현장에 적용한 결과를 보도록 하자.

먼저, 설계 CAD 서버에서는 강재에 표시할 문자열(부재 정보)을 사전에 생성하여 입력하고 해당 강재가 컨베이어에 들어오면 Fig. 8 과 같이 강재 정보가 입력 된다.

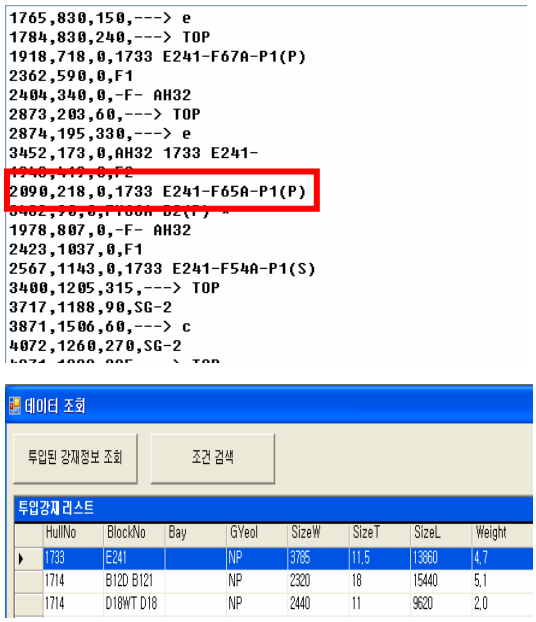


Fig. 8 Characters (above) & Steel plate (below) Information

Fig. 8 의 문자열 정보에는 텍스트 파일 형식으로 한 줄마다 순서대로 문자열이 출력될 위치(X, Y 좌표), 각도, 문자열이 포함되어 있다. 강재 정보에는 강재의 사용처 및 강재의 규격 등이 포함되어 있다.

상기 두 가지 정보를 조합하여 Fig. 9 와 같이 강재에 출력할 데이터를 생성한다.

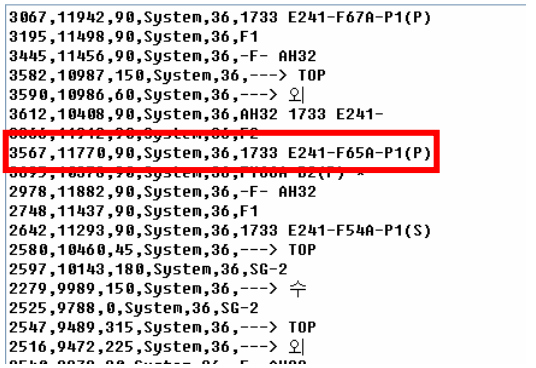


Fig. 9 Printing data file before compensation

강재 출력 데이터에는 문자열 데이터에 추가적으로 글자 폰트의 종류 및 크기 등을 설정할 수 있다. 그 외에도 잉크 헤드의 동작과 관련된 정보를 포함하고 있다.

운영부에서는 계측부에서 전송한 계측 데이터를 이용하여 강재 출력 데이터의 좌표 및 각도를 보정하고 문자열 그룹 분류 규칙에 의거해서 몇 번째 블록에 출력할 것인지 결정한다. Fig. 10 은 블록 결정 및 좌표 보정이 적용된 강재 출력 데이터이다.

```
12,3217,11300,90,System,36,1733 E241-F54A-P1(S)
12,3324,11408,90,System,36,1
12,4145,11409,90,System,36,(P)
12,4163,10987,60,System,36,---> 외
13,1090,12240,90,System,36,F1
13,3324,11443,90,System,36,F
13,3535,12081,330,System,36,---> TOP
13,3535,12100,240,System,36,---> 외
13,3557,11887,90,System,36,-F- AH32
13,3646,11946,90,System,36,1733 E241-F67A-P1(P)
13,3771,11501,90,System,36,F1
13,3945,11914,90,System,36,F2
13,4145,11771,90,System,36,1733 E241-F65A-P1
14,1978,13051,120,System,36,AH32 1733 E241-B201(S)
```

Fig. 10 Printing data file after compensation

Fig. 10 에서 맨 처음 두 자리의 숫자는 해당 문자열이 몇 번째 블록이 출력될 것 인지를 나타낸다. Fig. 9, 10 을 서로 비교해 보면 원래 같은 문자열에 포함된 글자가 분할되어 다른 블록에 들어간 것을 알 수 있다.

이 후, 해당 강재 출력 데이터 파일은 강재 출력 이미지 파일로 변환되어 잉크 헤드로 전송된다. Fig. 11 은 상기 강재 출력 이미지 파일의 12 번째 블록과 13 번째 블록의 일부를 시계 방향으로 90 도 회전 후 확대한 것이다.

Fig. 11 Printing Image files of 12th and 13th block

Fig. 11 을 보면 잉크 헤드가 좌우 왕복 운동을 하며 출력 시작 위치가 바뀌기 때문에 이미지가 좌우로 반전되어 있다.

마지막으로 Fig. 12 는 Fig. 11 의 이미지 파일이 실제 강판에 출력된 결과를 보여준다.

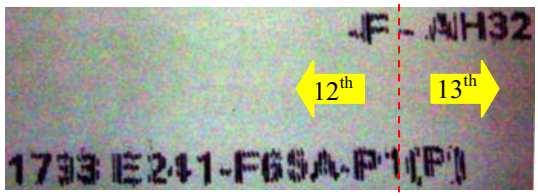


Fig. 12 Printed characters which are split into 2 blocks

## 5. 결론

본 연구에서는 이동식 자동 마킹 시스템에서 필수 불가결한 요소인 문자 분할 알고리즘에 대해 연구하였다. 이동식 자동 마킹 시스템은 문자열을 블록 단위로 출력하는데 문자열이 여러 개의 블록에 걸쳐 출력되는 경우 문자를 분할하지 않고 출력하게 되면 문자에 단차(잘림) 현상이 나타나게 된다.

그러므로 본 논문에서는 문자열을 위치에 따라 6 개의 그룹으로 분류하고 각 그룹 별로 문자열의 분할, 재배치 및 새로운 좌표를 계산하는 방법을 고안했다. 그리고 본 알고리즘은 Microsoft 사의 .NET Framework 기술(Sells and Gehtland 2004, Mahesh 2004, Anthony and Jim 2002)을 기반으로 하여 시스템을 구현하였으며 실제 현장에 적용하여 목적한 성과를 확인하였다.

그리고 앞으로는 문자열의 출력 품질 개선 문제에 초점을 두어 연구를 수행할 예정이다.

## 참 고 문 헌

• Ahn, J.G., 2007, “ Printing Robot System for labeling,” Final Report of Samsung Heavy INDUSTRIES CO.,LTD, Vol. 1, No. 1, pp. 1-99.

- Ryu, S.W., Back, S.C., Park, D.K., Bae, S.J., Lee, D.H. and Ahn, J.G., 2008, " Printing Robot System for Labeling on the Moving Steel Plate," International Conference on Control Automation and Systems, pp. 1484-1489.
- Sells, C. and Gehtland, J., 2004, " Windows Forms Programming in Visual Basic.NET," Addison Wesley, pp. 1-680.
- Mahesh, C., 2004, " Graphics Programming with GDI+," Addison Wesley, pp. 1-740.
- Anthony, J. and Jim, O., 2002, Network Programming with Microsoft Windows 2nd Edition," Microsoft Press, pp. 1-580.

< 박 두 경 >

< 한 상 동 >

< 김 호 구 >

< 안 정 기 >

< 김 대 경 >