

특징점 매칭을 이용한 가공철근 꼬임 판단 알고리즘

박종찬* 강대성**

An Algorithm for the Determination of Twisted Rebar using Feature Matching

Jong-Chan Park*, Dae-Seong Kang**

이 논문은 2017 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No.2017R1D1A1B04030870)

요 약

최근 기업들은 AI 영상 분석 기반의 스마트 팩토리를 도입하여 가공철근 공장의 핵심설비인 절단/절곡, 적재 자동화 설비를 개선함으로써 가공철근 생산성을 향상시키고 있다. 본 연구는 가공철근의 실시간 철근 꼬임 검출에 대한 연구이며, 철근이 사출되는 시작지점부터 철근이 절단되기 직전까지의 철근의 꼬임을 판단을 해야 하는 요구사항을 만족해야 한다. 철근 꼬임 정확성을 높이기 위해서는 철근 설비(배경)과 철근(객체) 분리가 중요한데 ORB 특징점 기반 검출방법을 통해서 환경에 따라 오검출 발생 확률이 높다. 본 논문에서는 환경에 따른 오검출을 줄이기 위해 ORB 알고리즘에 ROI, 허프 변환 직선 알고리즘을 적용한 검출 방법을 제안하며, 허프 변환 직선 알고리즘은 ROI영역 안에서 두 철근의 직선을 검출하는데 사용하였다. 제안하는 방법을 통해 연산 시간이 104msec/frame에서 54msec/frame으로 개선이 되어 약 50msec, 꼬임 검출 판단 정확도는 73%에서 98%로 개선이 되어 약 15%의 개선된 결과들을 얻었다.

Abstract

Companies have recently introduced smart factories based on AI image analysis to improve processing steel's productivity by improving the core facilities of processing reinforcement plants such as cutting and loading automation facilities. This research is a study on the detection difficulties of real-time rebar processing and must meet the requirements for determining the rebar kinks from the start of the rebar injection to just before the rebar is cut. In order to increase the accuracy of rebar twisting identification, the separation of rebar facilities (background) and rebar (objects) is essential, and the existing feature point-based detection methods have a high probability of misdetection depending on the environment such as illumination and dust. In this paper, we propose a detection method that applies ROI(Region Of Interest) and Hoffman straight algorithms to the ORB algorithm to reduce environmental misdetection, and Hoffman straight algorithms are used to detect straight lines of two rebars within the ROI region. The proposed method improves computational time from 104msec/frame to 54msec/frame, improving approximately 50msec and twisting detection judgment accuracy from 73% to 98%, resulting in improved results of approximately 15%.

Keywords

feature matching, line detection, region of interest, machine vision camera, rebar

* 동아대학교 전자공학과 석사과정
- ORCID: <https://orcid.org/0000-0002-1886-534X>
** 동아대학교 전자공학과 교수(교신저자)
- ORCID: <https://orcid.org/0000-0003-0186-2430>

· Received: Jan. 15, 2021, Revised: Feb. 19, 2021, Accepted: Feb. 22, 2021
· Corresponding Author: Dae-Seong Kang
Dept. of Dong-A University, 37 NaKdong-Daero 550, beon-gil saha-gu,
Busan, Korea,
Tel.: +82-51-200-7710, Email: dskang@dau.ac.kr

1. 서 론

최근 4차 산업혁명의 흐름 속에서 인공지능(AI), 딥러닝 등의 기술을 기반으로 머신러닝 기술이 빠르게 성장하고 있다. 특히 머신비전 중 산업용 카메라와 렌즈 관련 기술은 제품 불량 검출의 정확성을 높여 품질 향상에 기여함으로써 공장 무인 자동화에 필요한 다양한 연구가 진행 중이다[1].

가공철근 자동화 설비는 철근 자동절단/절곡 공정에서 불량 제품 생산 시 교정 작업 시간과 교정 정확도는 작업자의 숙련도에 100%에 의존하는 부분들을 손실률을 최소화하는 자동생산 프로그램과 로봇 제어가 연계되어 통합 자동화된 스마트 생산시스템을 목표로 한다. 즉, 스마트 팩토리 분야에 AI 영상분석기술 적용 확대하는 것이 최종 목적이다[1].

본 연구는 가공철근의 실시간 철근 꼬임 검출에 대한 연구이며, 철근이 사출되는 시작지점부터 철근이 절단되기 직전까지의 철근의 꼬임을 판단해야 하는 요구사항을 만족해야 한다. 본 논문에서는 환경에 따른 오검출을 줄이기 위해 ORB(Oriented FAST and Rotated BRIEF) 알고리즘에 ROI(Region Of Interest), 허프 변환(Hough-transform) 직선 알고리즘을 적용한 검출 방법을 제안하며, 허프 변환 직선 알고리즘은 ROI 영역 안에서 두 철근의 직선을 검출하는데 사용하였다. 제안하는 방법을 통해 연산 시간이 104msec/frame에서 54msec/frame으로 개선이 되어 약 50msec, 꼬임 검출 판단 정확도는 73%에서 98%로 개선이 되어 약 15%의 개선된 결과들을 얻었다.

본 논문의 구성은 객체 검출 관련 이론, 제안하는 방식, 실험 방법 및 성능평가, 결론 및 향후 과제에 대한 설명으로 끝이 난다.

II. 객체 검출 관련 이론

철근의 영상을 획득하기 위해 가공철근 사출구로부터 일정 거리에 떨어진 위치에 카메라를 상부에 설치하고, 이를 기반으로 꼬임 판단을 한다. 철근의 꼬임을 판단하는 세부 기술 알고리즘들은 아래와 같은 방법이 존재한다.

2.1 ORB 특징점 검출

FAST(Features from Accelerated Segment Test) 알고리즘은 아래 그림 1과 같이 중앙의 화소와 이웃 화소와의 차이의 임계값, threshold 하나만 인수로 가진다. 코너점의 정도를 판단하는 수단이 없어, FAST에 의한 특징점을 정렬할 수단이 없다. ORB는 다중 스케일 피라미드의 각 단계에서 Harris 코너 반응 값을 사용하여 필터 된 특징점을 계산한다.

특징점의 방향을 패치에서 밝기의 모멘트를 사용하여 $\theta = a \cdot \tan^2(m_{01}, m_{10})$ 로 단순하게 계산한다. 코너점(O)의 밝기가 모멘트 계산에 의한 밝기의 중심 $C = (m_{10}/m_{00}, m_{01}/m_{00})$ 으로부터 떨어져, O에서 C로의 벡터 방향을 특징점의 방향으로 계산한다. ORB는 특징점을 검출하기 위하여 이웃을 결정하는 반지름이 9인 FAST-9를 사용한다.

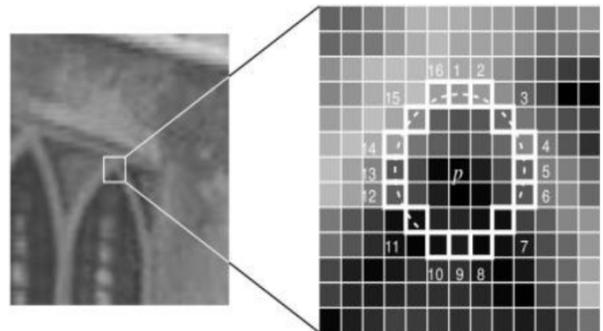


그림 1. FAST 특징점 검출의 예
Fig. 1. Example of FAST feature detector

2.2 ORB 디스크립터(Descriptor) 계산

ORB는 특징점의 방향에 따라 BRIEF(Binary Robust Independent Elementary Features)알고리즘을 조정하여 계산하는 방법을 사용한다. 패치 P에서, n개의 (x, y) 위치 쌍에서, 위치 $(x_i, y_i), i = 1, \dots, n$ 에서, n개의 이진 테스트에 의한 이진 비트열을 생성하기 위한 $2 \times n$ 행렬 $S = (x_i, y_i)^T, i = 1, \dots, n$ 을 정의한다. 패치의 회전각도 θ 에 대응하는 회전행렬 R_θ 를 사용하여, S의 회전 조정 버전 $S_\theta = R_\theta S$ 을 계산한다. 회전각도 θ 에 의한 특징 집합 S의 n-차원의 이진 비트 스트링, $g_n(p, \theta)$ 을 계산한다.

$$g_n(p, \theta) = f_n(p) \mid (x_i, y_i) \in S_\theta \quad (1)$$

$$\text{여기서, } f_n(p) = \sum_{1 \leq i \leq nd} 2^{i-1} \tau(p; x_i, y_i) \quad (2)$$

$$\tau(p; x, y) = \begin{cases} 1, & \text{if } p(x) < p(y) \\ 0 & \text{o.w} \end{cases} \quad (3)$$

회전각도 θ 는 특징점 검출에서 계산한 값이다. ORB 논문에서는 회전각도 θ 를 12도 간격으로 미리 계산된 BRIEF 패턴에서 참조표를 구성해 놓고 사용한다.

300k의 특징점 집합에서, 31×31의 패치의 각 이진 테스트를 5×5 윈도우 쌍으로 할 때, $N = (31-5)^2$ 개의 윈도우가 가능하고, 모든 가능한 N개에서 2개를 뽑는 이진 테스트 조합에서, 겹치는 영역을 제거하면 $M=205,590$ 개의 테스트가 가능하고, 이들 각 테스트에 대하여 가우시안 평균이 0.5 근처인 상관관계가 없는 $n=256$ 개의 테스트 집합을 Greedy 탐색으로 사용하여 찾는다[2][3].

2.3 ROI 지정

이미지의 특정 영역을 지정하는 것에 대해 ROI라고 한다. 영상에서 먼저 원하는 지역의 좌측 상단에 해당하는 좌표값과, 지정하고자 하는 영역의 폭과 높이를 각각 지정합니다. 그 후 ROI에 영상의 해당 영역을 설정하고, 테두리 선을 그려줍니다. 적용 시 이미지에는 아래 그림 2와 같이 테두리 선이 나타나게 됩니다[4].

철근을 외 배경을 제거하기 위해 차영상 방식을 시도하여 배경을 제거하였으나, 조명의 변화에 따라서 잡음이 많이 발생하는 문제점이 발생하였다.

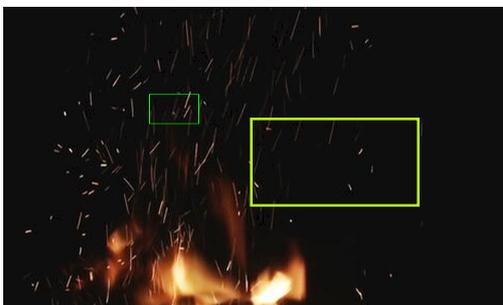


그림 2. 관심영역 지정의 예
Fig. 2. Example of ROI selection

2.4 허프 변환 직선 검출

허프 변환은 이미지에서 직선을 찾는 가장 보편적인 알고리즘이다. 아래 그림 3과 같이 이미지에서 선과 같은 단순한 형태를 빠르게 검출할 수 있다. 허프 변환은 이미지 내의 어떤 점이라도 선 집합의 일부일 수 있다는 가정하에 직선의 방정식을 이용해 직선을 검출한다. 직선 검출은 직선의 방정식을 활용해 $y = ax+b$ 를 극좌표(ρ, θ)의 점으로 변환해서 사용한다.

극좌표 방정식으로 변환한다면 $p = x \sin \theta + y \cos \theta$ 이 되어, 직선과 원점의 거리(ρ)와 직선과 x축이 이루는 각도(θ)를 구할 수 있다. 영상에서 직선을 검출하기 위해서, 전처리 작업을 진행해야 한다. 전처리를 진행하기 위해 그레이스케일과 캐니 엣지 알고리즘을 사용한다[5][6].

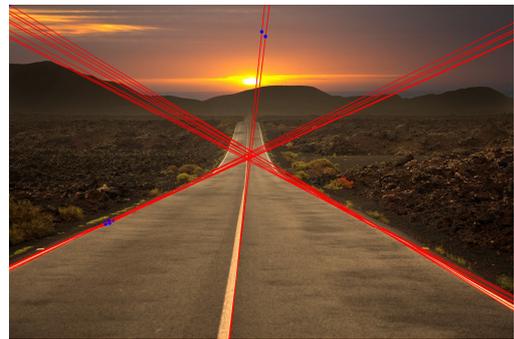


그림 3. 허프 변환 직선 검출 결과
Fig. 3. Straight line detection result using Hough-transform

III. 제안하는 방식

제안한 특징점 매칭을 이용한 가공철근 꼬임 판단 알고리즘은 크게 ORB 알고리즘을 이용해서 특징점을 추출, 특징점을 추출한 뒤에 디스크립터에 정보를 저장, 기존의 저장된 디스크립터들과 비교하여 매칭점을 찾은 뒤 디스크립터 정보와 매칭 비교하여 정상으로 판별된 경우 pass, 철근이 꼬였을 경우 twist라고 판단하는 부분으로 구성된다[7]. 여기서 가공철근의 연산시간, 정확도, 환경에 강인하기 위해 카메라 영상 내의 철근이 나오는 부분에 ROI 적용을 한다. 다음 그림 4는 특징 데이터 기반 철근 꼬임 판단 프로세서이다.

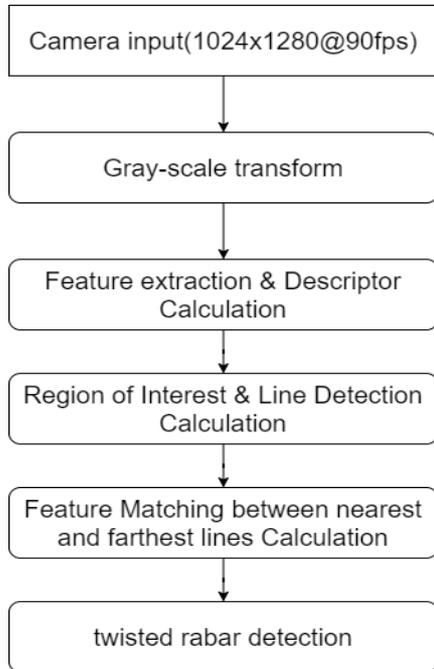


그림 4. 특징 데이터 기반 철근 꼬임 판단 프로세서
Fig. 4. Feature data based twisted rebar detection process

ROI 영역 안에서 허프 변환 직선 검출 알고리즘을 적용하여 철근의 직선을 검출하면 불필요한 직선들을 여러 개 검출하게 된다. 이를 해결하기 위해 ROI 테두리 중심 상단 좌표에 점을 찍은 후 점과 허프 변환 직선 검출을 통하여 나온 직선과의 거리 방정식을 정립하여 가장 가까운 직선과 가장 멀리 있는 직선 철근의 두 직선을 구한 뒤에 두 직선 사이에 ORB 알고리즘을 통하여 두 직선 사이의 특징점을 매칭 함으로써 연산 시간, 정확도, 환경에 강인한 제안하는 방법을 제시한다.

머신 비전 카메라의 노출 시간은 6000, 프레임은 90으로 설정하였고 ORB 검출기의 임계값 변화에 따른 특징점 추출 결과는 그림 5, 6, 7과 같다. 임계값(300) 미만으로 낮은 그림 5와 같이 철근을 포함한 철근이 나오는 사출 부분의 특징점이 추출된다. 그리고 임계값(350)을 설정하면 그림 6과 같이 상단 철근의 특징점이 검출된다. 또한, 그림 6에 설정된 임계값을 이 이상으로 높일 경우 그림 7과 같이 특징점이 검출되지 않는다.

실제 그레이 영상에서 특징점을 검출하는 ORB 검출기 임계값을 150으로 설정하여 검출되는 특징점들을 철근으로 정의한다.

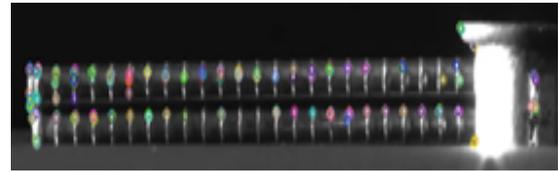


그림 5. 임계값 = 150
Fig. 5. Threshold value= 150

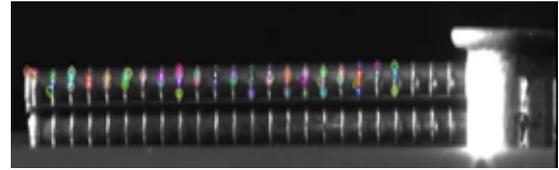


그림 6. 임계값 = 350
Fig. 6. Threshold value= 350

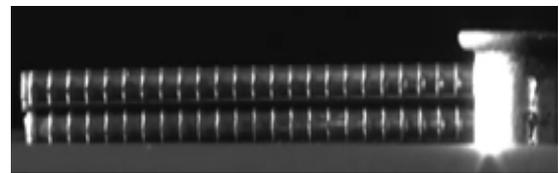


그림 7. 임계값 = 400
Fig. 7. Threshold value= 400

기존의 ORB 알고리즘을 통해서 꼬임 검출을 하는 과정은 특징점을 추출한 뒤에 디스크립터에 정보를 저장, 기존의 저장된 디스크립터들과 비교하여 매칭점을 찾은 뒤 디스크립터 정보와 매칭 비교하여 정상으로 판별된 경우 pass, 철근이 꼬였을 경우 twist라고 판단하는 결과는 그림 8, 9와 같다.

철근 두 줄이 일정하게 나오면 그림 8과 같이 pass라고 표시하는 걸 볼 수 있다.

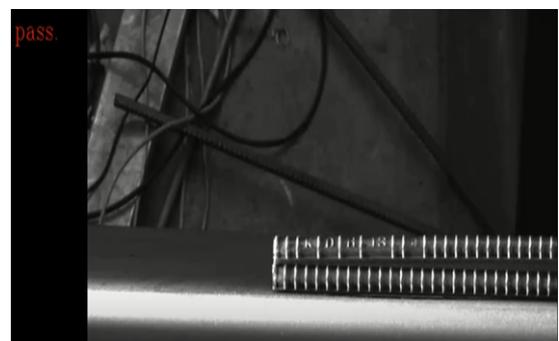


그림 8. ORB 실시간 검출 “pass”
Fig. 8. Real-time detection after ORB application “pass”

철근이 왼쪽으로 가거나 오른쪽으로 꼬인 경우 그림 9와 같이 twist라고 표시하는 걸 볼 수 있다.

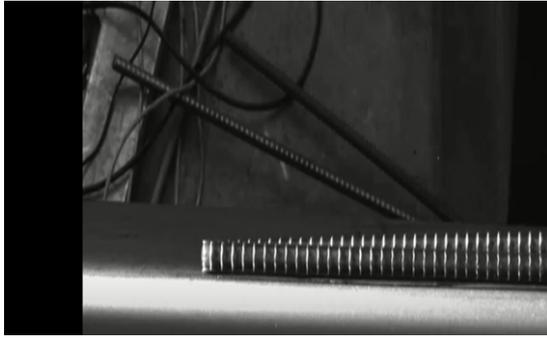


그림 9. ORB 실시간 검출 "twist"

Fig. 9. Real-time detection after ORB application "twist"

ROI 영역은 시작 좌표(x,y), 영역의 폭(w), 높이(h)라고 하면 영상의 y행에서 y+h행까지, x열에서 x+w열까지 슬라이싱을 통해 영역을 지정하게 된다. 실시간 영상이 1024(폭)x1280(높이)이고 철근이 영상의 중심으로 지나가므로 중심을 기점으로 $x=0$, $y=360$, $w=1280$, $h=180$ 좌표설정 한 뒤에 $y=y+h$, $x=x+w$ 으로 ROI영역 지정을 하였다.

제안된 방법은 가공철근의 검출속도와 꼬임 판단의 정확성을 높이기 위해 카메라 영상 내의 ROI영역을 지정하여 철근이 나오는 부분만 매칭을 하게 한 결과는 그림 10과 같다.

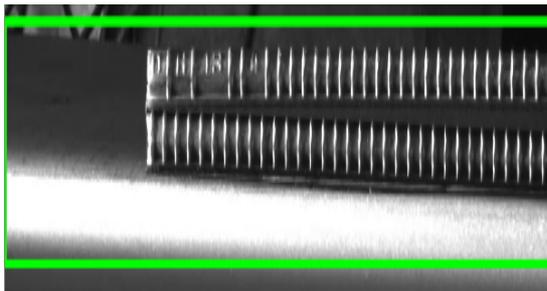


그림 10. ROI 지정

Fig. 10. Selection of ROI

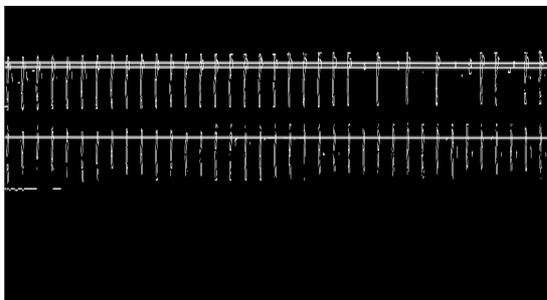


그림 11. ROI 내에서 허프 변환 직선 검출

Fig. 11. Straight line detection using Hough-transform in ROI

ROI 영역 안에서 허프 변환 직선 검출을 한 경우 그림 11과 같이 불필요한 직선들이 검출되는 것을 확인할 수 있다.

ROI영역 안에서 불필요한 직선을 없애고 철근의 가장 가까운 직선과 멀리 있는 두 직선을 검출하기 위해 허프 변환 직선 검출을 알고리즘에 거리 방정식을 통하여 그림 12와 같이 두 직선의 검출을 확인할 수 있다.

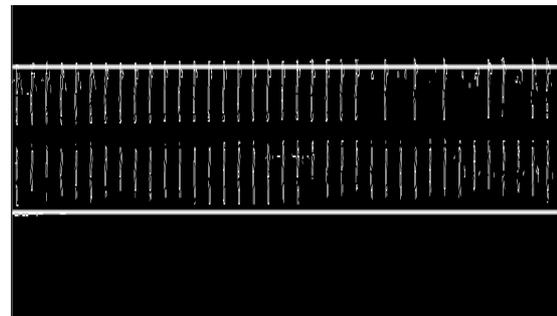


그림 12. 거리 방정식을 통한 두 직선 검출

Fig. 12. Detection of two straight lines through distance equation

최종적으로 가공철근의 꼬임을 검출할 때 복잡한 환경에서 불필요한 잡음을 줄이기 위해 ROI 영역을 지정한 뒤에 그림 13과 같이 ROI 테두리 중심 상단 좌표에 점을 찍은 후 점과 허프 변환 직선 검출을 통하여 나온 직선과의 거리 방정식을 정립한다. 이후, 가장 가까운 직선과 가장 멀리 있는 직선 철근의 두 직선을 구한 뒤에 두 직선 사이에 ORB 알고리즘을 통하여 두 직선 사이의 특징점을 매칭을 한 결과를 확인할 수 있다.

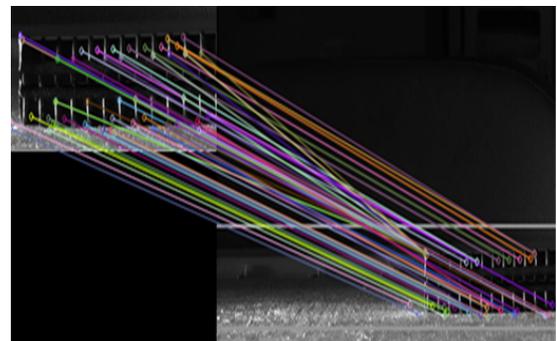


그림 13. 검출된 두 직선사이의 특징점 매칭

Fig. 13. Matching of feature points between two detected straight lines

제안하는 방식을 통해 그림 14, 15와 같이 실시간으로 철근의 꼬임 유무를 실시간으로 검출하였다.

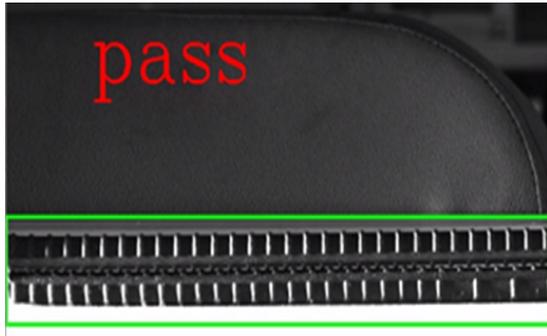


그림 14. ROI, 허프 변환 직선 검출 적용 후 실시간 검출 “pass”

Fig. 14. Real-time detection after ROI, line detection application “pass”



그림 15. ROI, 허프 변환 직선 검출 적용 후 실시간 검출 “twist”

Fig. 15. Real-time detection after ROI, line detection application “twist”

IV. 실험 방법 및 성능평가

본 실험의 소프트웨어 환경은 윈도우 10에서 주피터 노트북(Jupyter Notebook)으로 파이썬(Python) 언어를 사용하며, Open-CV의 ORB 알고리즘을 이용하여 실험하였다.

실험환경은 그림 16의 MEP사의 절곡기를 통해서 진행하였다. 실험에 사용한 카메라는 그림 17의 HIKVISION MV-CAD13-20GM Machine Vision 카메라를 이용하여 사출되는 부분의 상부에서 사출되는 실시간 영상을 1024x1280@90fps으로 설정하고 이 영상을 기반으로 꼬임 검출에 대해서 위 제안 방식을 토대로 결과값을 도출하였다[8][9].

제안하는 알고리즘의 성능 평가하기 위해 기본 ORB 알고리즘을 이용하여 철근을 검출하는 연산 시간과 제안된 방법으로 검출하는 연산 시간의 평균과 꼬임 판단 정확도를 비교하였다.

Video 1은 환경에 영향을 받지 않았을 때의 영상이며, Video 2의 경우 외부 환경의 강한 조명에 의해 철근의 아래에 조명이 반사되었을 때의 영상이다.



그림 16. MEP사의 가공철근 절곡기
Fig. 16. MEP's fabricated steel curves



그림 17. HIKVISION MV-CAD13-20GM 머신 비전 카메라
Fig. 17. HIKVISION MV-CAD13-20GM machine vision camera

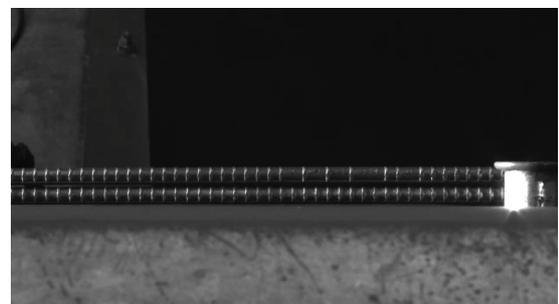


그림 18. 비디오 1
Fig. 18. Video 1

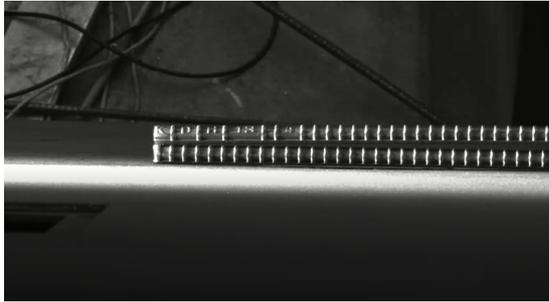


그림 19. 비디오 2
Fig. 19. Video 2

표 1은 철근 꼬임 유무 판단 시간 측정을 나타낸 결과이다. ORB만 사용하여 철근 꼬임을 판단했을 때 그림 18에서는 0.104 sec/frame의 결과가 그림 19에서는 0.143 sec/frame 결과를 얻었다. 그림 18에서는 조명이 강하여 연산 시간이 증가하였다. 제안된 방법에서의 그림 18에서는 0.054 sec/frame의 결과 그림 19에서는 0.064 sec/frame의 결과를 얻었다.

표 1. 철근 검출 평균 연산 시간 비교(sec/frame)
Table 1. Comparison of average operation times for rebar detection(sec/frame)

| | ORB | Proposed algorithm |
|---------|----------------|--------------------|
| Video 1 | 0.104sec/frame | 0.054sec/frame |
| Video 2 | 0.143sec/frame | 0.064sec/frame |

표 2는 철근 꼬임 판단의 정확도를 나타낸 결과이다. 꼬임 영상 데이터 100개를 통하여 꼬임 판단의 정확도를 나타낸 결과이다. “정확도(%) = 올바르게 판단한 영상 개수 / 전체 영상 개수”를 통하여 정확도를 측정하였다. ORB만 사용하여 철근 꼬임을 판단했을 때 그림 17에서는 73% 결과가 그림 18에서는 65%의 정확도를 보였다. 제안된 방법에서의 그림 18에서는 98%의 결과 그림 19에서는 95%의 정확도 결과를 얻었다.

표 2. 꼬임 판단 정확도 비교(정확도(%))
Table 2. Compare twisted decision accuracy (accuracy(%))

| | ORB | Proposed algorithm |
|---------|-----|--------------------|
| Video 1 | 73% | 98% |
| Video 2 | 65% | 95% |

제안된 방법이 ORB 알고리즘만을 적용했을 때보다 연산시간과 정확도가 우수한 성능을 내는 이유는 ORB 알고리즘만을 적용하였을 때는 영상 전체의 특징점들을 추출하고 매칭을 해야 하며 또한 환경의 변화에 의해서 불필요한 특징점 정보들을 추출하고 매칭을 하게 되면 연산 시간과 매칭의 정확성의 성능이 떨어지게 된다.

그러므로 제안하는 방법을 통해 ROI 영역 지정 후에 거리방정식을 적용한 허프 변환 직선 검출을 통해 ORB 알고리즘을 적용할 경우 오직 두 철근만의 특징점을 추출하여 데이터를 최소화하여 연산이 개선되고, 정확한 특징점 데이터들의 매칭을 통해 정확도가 증가하여 환경의 변화에도 강인한 우수한 성능을 낼 수 있다.

V. 결론 및 향후 과제

ORB 알고리즘은 FAST 알고리즘과 BRIEF 알고리즘이 합쳐진 알고리즘이다. 특징 검출 알고리즘으로는 SIFT와 SURF가 있지만 연산량이 많아 실시간으로 물체를 인식하는데 어려움이 있어 특징점 추출이 빠른 ORB를 사용하였다. 하지만 ORB 알고리즘은 특징점이 많아질수록 객체를 인식하는 연산속도가 느려지는 단점이 있다.

본 논문에서는 이러한 단점을 보완하기 위해 카메라 영상 내의 철근이 나오는 ROI 영역 설정 후 ROI 테두리 중심 상단 좌표에 점을 찍은 후 점과 허프 변환 직선 검출을 통하여 나온 직선과의 거리방정식을 정립하여 가장 가까운 직선과 가장 멀리 있는 직선 철근의 두 직선을 구한 뒤에 두 직선 사이에 ORB 알고리즘을 통하여 두 직선 사이의 특징점을 매칭 함으로써 꼬임의 꼬임 유무를 판단한다.

그 결과 ORB 알고리즘에 비해서 연산 시간이 104msec/frame에서 54msec/frame으로 개선이 되어 약 50msec, 꼬임 판단 정확도는 73%에서 98%으로 약 15% 정확도가 개선된 결과를 얻었다. 향후 연구는 두 직선을 이용하여 두 철근의 기울기를 계산하여 철근의 IN/OUT 꼬임 구분을 하는 것이 추후 연구의 목표이다.

References

- [1] J. S Kim and D. S Kang, "Real-time Rebar Injection Endpoints Tracking Method to Improve the Straightness of Rebar", Journal of KIIT, Vol. 17, No. 8, pp. 75-83, Aug. 2019.
- [2] G. H Lee, S. W Lee, and Y. G S, "Improved object recognition results using SIFT and ORB feature detector", Journal of Digital Contents Society, Vol. 22, No. 1, pp. 151-158, Jan. 2021.
- [3] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: an efficient alternative to SIFT or SURF", IEEE, Vol. 1, No. 1, pp. 2564-2571, Nov. 2011.
- [4] Python OpencvCV Line Detection, Dae Hee Yun, <https://076923.github.io/posts/Python-opencv-28>, [accessed: Feb. 23. 2020]
- [5] [OpenCV] Setting an area of interest in an image, Doony, <https://hyongdoc.tistory.com/355>, [accessed: Feb. 3. 2020]
- [6] L. Ding and A. Goshtasby, "On the Canny edge detector", Pattern Recognition, Vol. 34, No. 3, pp. 721-725, Mar 2001.
- [7] W. J Moon, I. J Yoo, Y. H Seo, and D. W Kim "Scene Change Detection and Filtering Technology Using SIFT", JBE, Vol. 24, No. 6, pp. 939-947 , Nov. 2019.
- [8] J. M Lee and D. S Kang, "Start Point Detection Method for Tracing the Injection Path of Steel Rebars", Journal of KIIT, Vol. 17, No. 6, pp. 9-16, Jun. 2019.
- [9] A. Barsamian, V. H. Berk, and G. Cybenko, "Target Tracking and Localization Using Infrared Video Imagery", SPIE Unattended Ground Sea and Air Sensor Technologies and Applications, Vol. 6231, pp. 1-7, May 2006.
- [10] J. S Kim, D. C Jung, and Y. K Kim, "Location based service, Bluetooth4.0, Measurement result filtering, Distance measurement correction algorithm, LED", JIIBC, Vol. 17, No. 2, pp

63-68, Apr 2017.

- [11] K. T Yang, B. K Lee, and Y. K Kim, "Media Facade, Matching process, Projection, Surface, Target, IR cameras", JIIBC, Vol. 18, No. 3, pp 9-16, Jun. 2018

저자소개

박 종 찬 (Jong-chan Park)



2020년 2월 : 동아대학교
전자공학과(공학사)
2020년 3월 ~ 현재 : 동아대학교
전자공학과 석사과정
관심분야 : 영상처리, 인공지능

강 대 성 (Dae-Seong Kang)



1994년 5월 : Texas A&M 대학교
전자공학과(공학박사)
1995년 ~ 현재 : 동아대학교
전자공학과(교수)
관심분야 : 영상처리, 패턴인식