YOLO CCB-OCC

개요

딥러닝은 수많은 분야에서의 적용 사례가 나오고 있으며, 이전까지는 잘 설계된 알고리즘에만 의존했던 문제에 대해서도 더욱 개선된 성능을 나타내고 있어 많은 이목을 끌고 있는 분야이다. 본 연구에서는 선행 연구인 Complementary Color Barcode-Based Optical Camera Communications(CCB-OCC) 시스템과 딥러닝 기반의 객체 인식 기법인 YOLOv3를 결합하여, 이전 연구보다 더 향상된 CCB-OCC 시스템에 대해 제안하고자 한다. 실험을 통해, 딥러닝 기반의 객체 탐지를 활용한 CCB-OCC가 파일럿 심볼 검출을 통한 동기화 과정에서의 알고리즘의 복잡도를 낮추고, 선행 연구보다 더 나은 수신율을 제공하여 수신 알고리즘을 간소화하고 성능을 확보한 CCB-OCC 시스템을 제공할 수 있음을 증명하였다.

소개

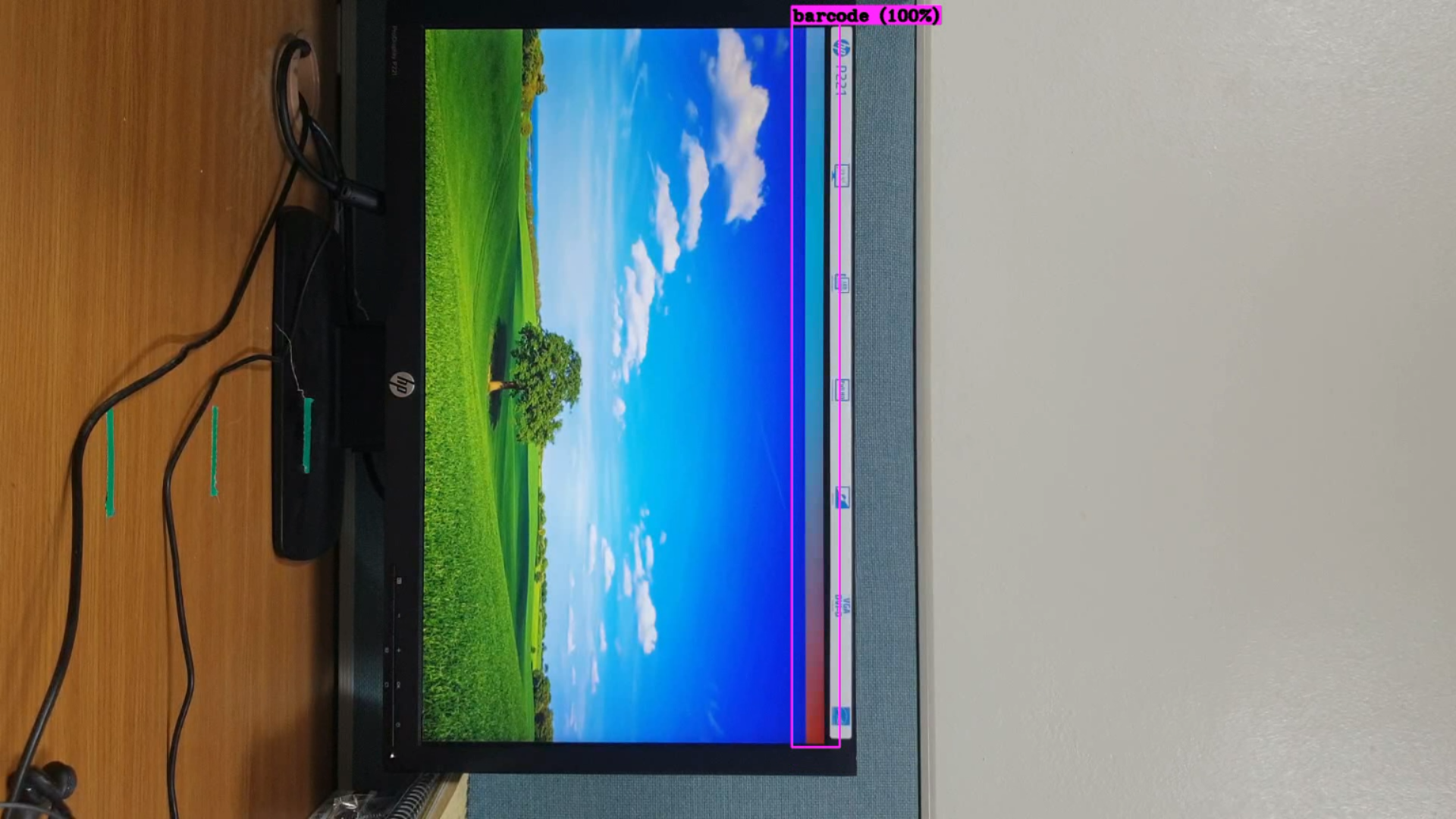
선행 연구에서 선보였던 CCB-OCC 기법은 비트열을 컬러 바코드 형태로 인코딩하고 이를 모니터의 디스플레이로 출력하고, 광학 카메라를 활용하여 컬러 바코드가 포함된 영상을 수신하는 구조를 가지고 있다. 선행 연구에서의 컬러 바코드 영역 추출 과정에서 딥러닝 기반의 실시간 객체 탐지 기법인 YOLO(You Only Look Once)를 사용하여 컬러 바코드 영역 추출 과정과 동기화를 위한 파일럿 프레임 탐지 과정에 적용하여, 기존 CCB-OCC 시스템의 수신 성능을 개선하고자 한다.

YOLO

YOLO(You Only Look Once)는 이미지를 SxS의 눈금으로 나누어, N개의 Bounding Box와 신뢰도를 예측한다. 여기서의 신뢰도는 Bounding Box의 정확성과 해당 구역에 실제 객체가 포함되는지에 대한 여부를 의미하며, 두 가지 값을 활용하여 예측된 Bounding Box 내에 해당 특정 클래스가 있을 확률에 대해 계산을 실시하는 방식으로 객체 탐지를 실시한다. 객체 탐지 분야에서 YOLO 등장 이전까지 주로 사용해왔던 Faster R-CNN(Region with Convolutional Neural Network)과 달리, 전체 이미지를 한 번에 보는 방식으로 빠른 속도로 객체 탐지가 가능하며 그 성능이 다른 객체 탐지 기법과 비교해볼 때, 크게 떨어지지 않는다는 것이 장점이다. 본 연구에서는 컬러 바코드의 영역을 탐지하고, 수신된 프레임에서 파일럿 프레임을 검출하기 위해 YOLO를 활용하였다.

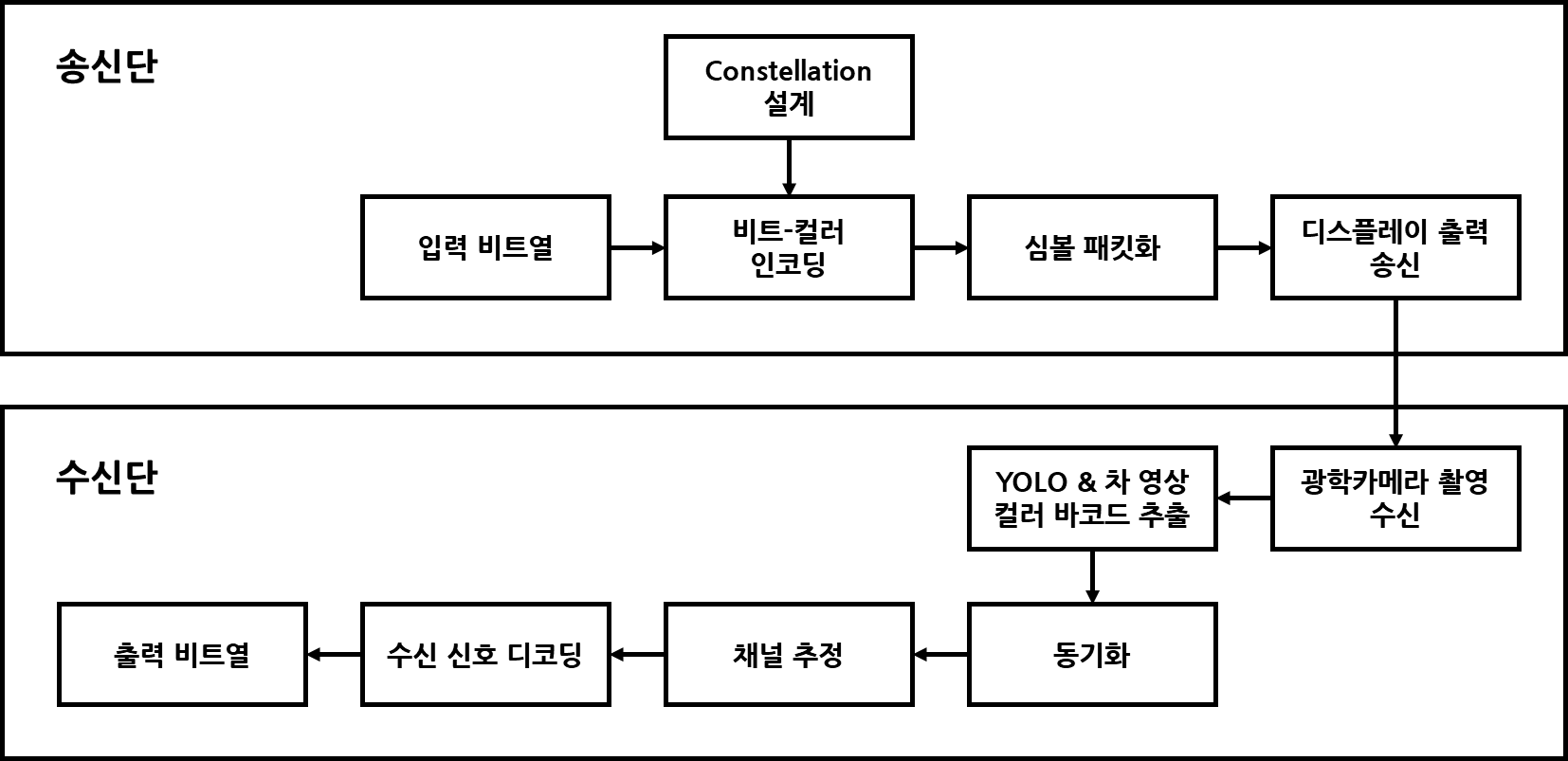
차 영상(Difference Image)

차 영상은 하나의 영상을 픽셀 단위로 다른 영상에서 뺀 영상을 의미한다. 이는 두 영상의 차이를 의미하며, 이전 영상과 다음 영상 사이의 변화를 탐지하는데 주로 사용된다. 본 연구에서는 YOLOv3를 활용한 컬러 바코드 탐지 과정에서 컬러 바코드 영역을 추출해내는 작업을 수행하였으나, 해당 영역에 컬러 바코드를 출력하는 디스플레이의 프레임이나 벽과 같은 배경 영역 등이 일부 포함되어 있어, 수신 신호 디코딩 과정에서 노이즈로 작용함을 확인하였다. 이는 채널 추정 과정이나 디코딩 과정에서 RGB value의 노이즈로 작용하여 정확한 데이터 수신이 이뤄지는데 있어 부정적인 영향을 미친다. 따라서, 추출 영역 내에서 실제 바코드 영역만을 추출하기 위해 차영상과 침식 및 팽창 연산을 적용하여 수신 영상에서 바코드 영역을 온전히 추출하여 수신 성능을 극대화하고자 하였다.

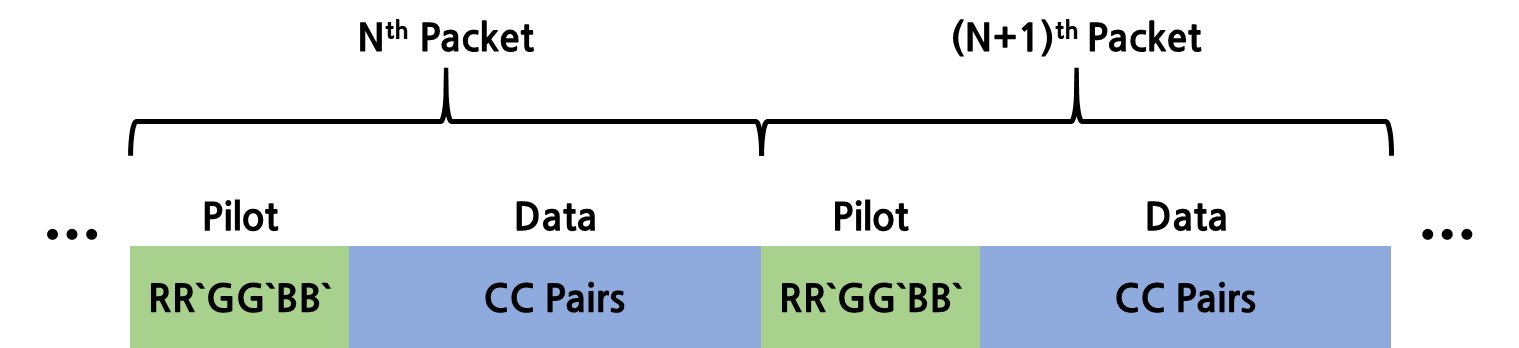


YOLO & Differential based CCB-OCC 시스템 구조

송신단에서는 데이터를 컬러 바코드로 인코딩한 후 이를 규정된 패킷으로 구성하여, 이를 디스플레이에 표시한다. 수신단에서는 롤링 셔터 방식의 광학 카메라를 활용하여 컬러 바코드를 포함한 동영상을 촬영하고, 촬영한 영상을 YOLOv3와 차영상을 이용해 컬러 바코드 영역을 추출하고 컬러 바코드의 구성에 대해 라벨링을 실시한 정보를 활용한다. 이렇게 추출한 컬러 바코드 영역과 라벨 정보를 활용하여 동기화와 채널 추정을 실시하고, 최종적으로 수신 신호 디코딩을 실시한다. 송신기와 수신기를 포함하는 CCB-OCC 시스템은 다음과 같이 구성되어 있다.

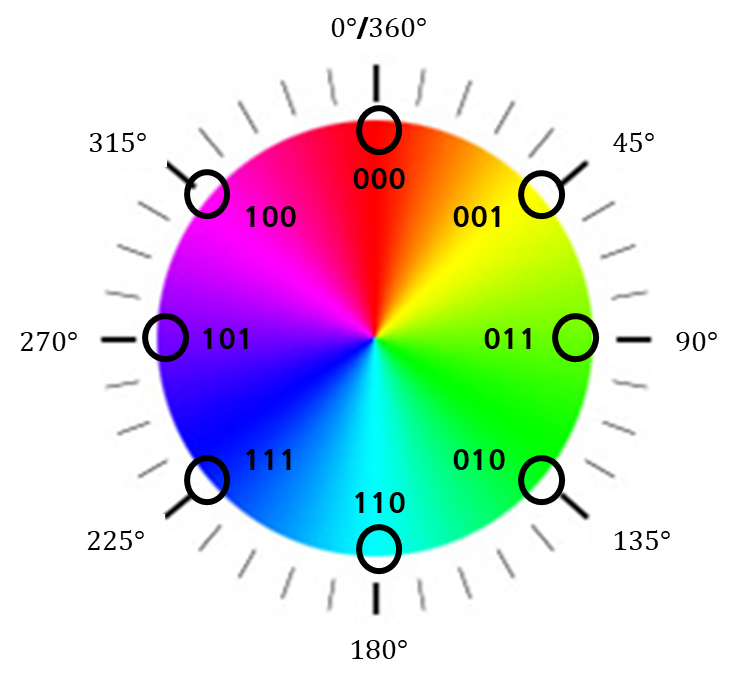


이 때 송수신 과정에서 컬러 바코드를 통해 출력되는 패킷과 심볼 구조는 선행 연구와 동일하게 구성되어 있다. 다음 도표는 선행 연구에서 활용한 파일럿 심볼과 데이터 심볼로 구성된 패킷 구조로 나타낸다.



패킷을 구성하는 심볼은 선행 연구와 동일한 심볼로 표현된다. 파일럿 심볼은 빛의 3원색인 빨강-초록-파랑(RGB)과 그 보색 관계(R`G`B)로 구성되며 데이터로 활용되는 데이터 심볼 역시 인코딩된 색상과 그 보색 관계의 색상으로 구성된다.

CCB-OCC에서는 심볼 인코딩을 위해서는 심볼에 대한 RGB 색상 값에 대해서 사전 정의가 필요하다. 본 연구에서는 선행 연구와 동일하게 Hue Color Constellation 상에서의 각도에 따라 각각의 심볼을 지정하는 방식을 사용하였다.



심볼 당 비트 수 역시 심볼 당 3bit로 지정하였고, 수신 심볼을 디코딩하는 과정에서 비트 에러를 최소화하기 위해 그레이 코딩 방식을 사용하여 Constellation 상에서 인접 심볼 간 비트의 차이가 1비트만 나타나도록 하였다. Hue Circle 상에서의 각도에 따른 RGB 색상은 다음과 같이 정의하였다.



동기화 및 채널 추정

D2C 통신에서의 패킷 동기화 과정은 영상을 통한 디코딩 과정에서 매우 중요하다. 선행 연구에서는 이를 위해 RR`GG`BB`으로 구성된 파일럿 프레임을 정의하였으며, 파일럿 프레임은 수신단에서 컬러 바코드 기반의 송신 영상을 촬영하는 시점에 따라, 연속적으로 3개 혹은 4개의 수신 프레임에 걸쳐 나타난다. 수신단에서 아래 도표와 같은 RR`GG`BB` 형태의 프레임을 연속적으로 수신하면 동기화 작업을 완료하고 채널 추정 작업을 실시한다. 선행 연구에서는 동기화를 위해 해당 파일럿 프레임을 검출하는 과정에서 수신된 컬러 바코드에 대해 복잡한 픽셀 계산과 함께, 송수신 장비의 노이즈에 대해 보상하는 작업을 수행해야 했다. 하지만, 본 연구에서는 YOLO v3를 활용하여 컬러 바코드 추정 영역 검출과 함께 파일럿 심볼 유무까지 동시에 추출할 수 있어 기존의 시스템과 다르게 수신단에서 파일럿 심볼 검출을 위한 복잡한 설계를 진행하지 않고 동기화를 진행할 수 있다.







채널 추정은 디스플레이의 영상 출력과 주변 광원, 수신을 시행하는 광학 카메라 등의 색상 오차로 인한 잡음에 대한 보상을 실시하기 위한 작업으로, CCB-OCC에서는 파일럿 심볼을 순수한 RGB로 정의하였기 때문에 이를 활용하여 채널 추정을 실시한다. 다음 도표는 RR`GG`BB` 형태로 수신된 파일럿 심볼에 대해 수신된 RGB 심볼을 추정하는 과정이다.

스크린샷이(가) 표시된 사진

자동 생성된 설명

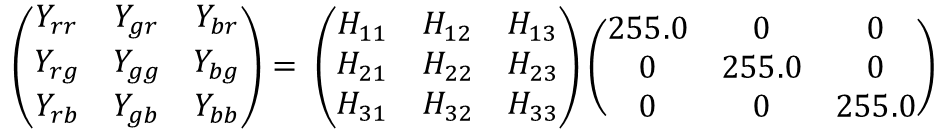
스크린샷이(가) 표시된 사진

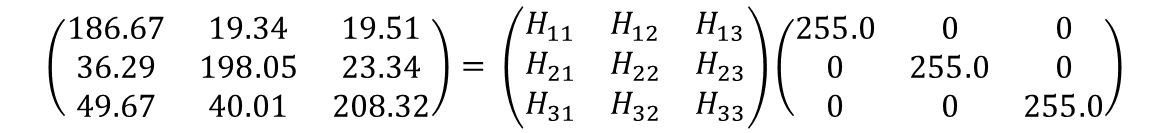
자동 생성된 설명

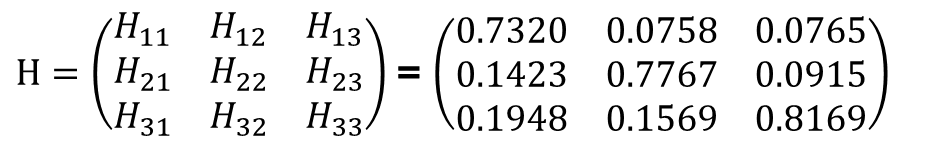
스크린샷, 지도이(가) 표시된 사진

자동 생성된 설명

이렇게 파일럿 심볼을 구성하는 3개 색상의 값을 획득하면 다음과 같은 3x3 형태의 정사각행렬을 구성할 수 있다. 선행 연구에서 제안된 것처럼 주변 환경과 장치의 특성에 따른 색상 잡음에 대해 보상해주는 채널 행렬을 다음과 같이 정의할 수 있다.





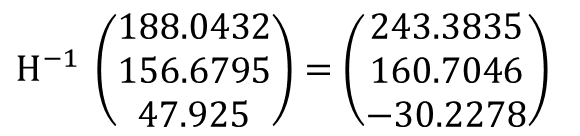


디코딩

디코딩 과정에서는 파일럿 심볼을 통한 동기화 과정 및 채널 추정 과정에서 획득할 수 있었던 수신 프레임 상 데이터 심볼의 위치를 통해 수신된 심볼의 RGB 값을 구하고 이를 채널 행렬의 역행렬과 곱하여 RGB 값에 대한 보상을 실시한다.

스크린샷이(가) 표시된 사진

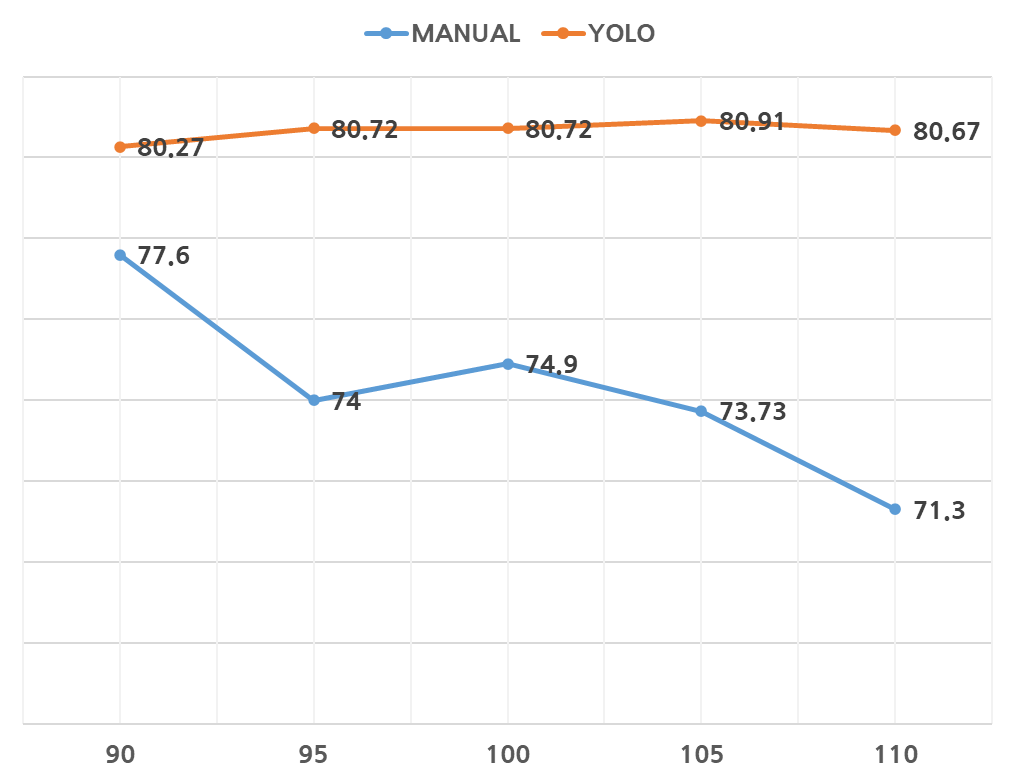
자동 생성된 설명



도표에서 확인할 수 있든 컬러 바코드의 252번째 픽셀의 위치가 데이터 심볼이 위치하는 것으로 추정하고, 해당 위치에서의 수신된 RGB의 값을 획득하였을 때 RGB의 값은 (188.0432, 156.6795, 47.925)를 나타낸다. 이렇게 획득한 RGB 값을 채널 행렬의 역행렬을 통해 보상을 실시한다. 이 때 획득한 RGB 색상을 Hue Circle의 각도로 나타내면 약 41.87도로, 이는 Hue Circle 상에서 45도에 해당하는 “001” 심볼로 판정하여 디코딩을 실시한다.

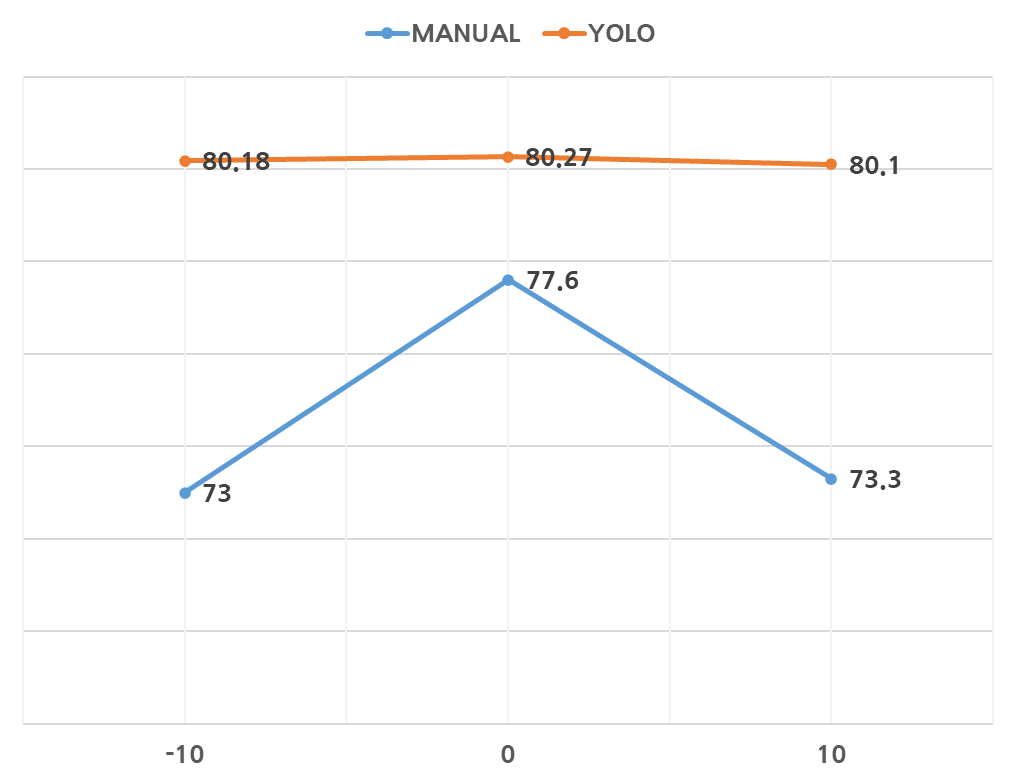
실험 및 평가

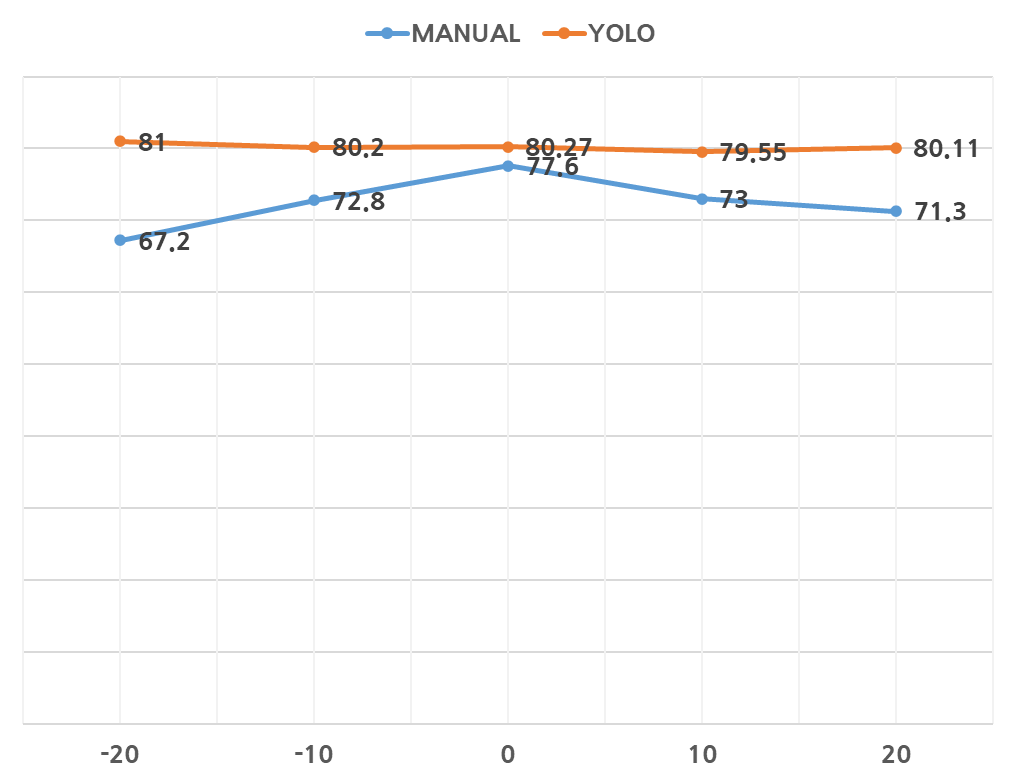
선행 연구와 비교하여 성능 개선 정도를 확인하기 위하여 수신단에서 YOLOv3와 차 영상을 적용한 CCB-OCC에 대한 실험을 진행하였다. 실험은 선행 연구와 동일한 방식으로 진행하였다. 수신 기기는 삼성 갤럭시 노트 9(Samsung Galaxy Note 9)의 후면 광학 카메라를 사용하였으며, 촬영 해상도와 촬영 프레임 속도는 1920x1080(FHD) 30fps로 설정하였고 이외의 세부 설정은 하드웨어 제조사인 삼성과 안드로이드 스마트폰에서의 기본 설정 값을 바탕으로 진행하였다. 송신 기기는 1920x1080(FHD) 60Hz의 모니터를 사용하였으며, 일반적인 실내 광원 환경에서 실험을 진행했다.



선행 연구와 동일하게 디스플레이와 카메라의 거리에 따른 평균 수신율을 측정했다. 카메라에 디스플레이가 전부 잡히는 최소 거리인 90cm부터 110cm까지 5cm씩 디스플레이와 카메라 사이의 거리를 늘려가며 진행했다. 수신 가능 비트율은 최대 80.91bps에서 최소 80.27bps로, 선행 연구와 비교해볼 때, 거리에 따른 데이터 수신율에 있어 성능 개선을 확인하였다.

추가적으로, 촬영 각도에 따른 변화를 수신 비트율을 확인하기 위한 실험을 진행하였다. 선행 연구에서의 실험 방식과 동일한 방식으로 디스플레이와 카메라 사이의 거리를 90cm로 고정하고, 상하각도와 좌우각도 변화에 따른 수신가능한 비트율을 측정도 진행하였다.





위 도표를 통해 알 수 있듯, 선행 연구와 비교해볼 때 상하 각도와 좌우 각도의 변화가 수신 가능한 비트율의 성능 저하가 나타나지 않는 것으로 나타났다. 따라서, 선행 연구에서 진행한 비교 실험에서 모든 경우에서 더 좋은 성능을 나타내어 기존의 CCB-OCC 시스템의 수신 성능을 조금 더 개선시켰다고 볼 수 있다.

결론

딥러닝 기반의 객체 탐지 기법인 YOLO v3를 활용한 CCB-OCC 시스템이 기존의 CCB-OCC 시스템보다 간소화된 수신단 알고리즘을 갖춘 상태에서도, 수신 성능이 우수한 것으로 나타났다. (다만, 이 논문에서의 극명한 성능 차이는 기존 논문에서는 차영상을 적용하지 않았기 때문에 상대적으로 YOLOv3를 패킷 동기화에 적용하고, 차영상을 활용한 본 연구와의 성능 차이가 있을 수 있다.)

본 연구에서는 YOLO v3를 활용하여 컬러 바코드 영역 추정과 함께 해당 수신 프레임에서의 파일럿 심볼 검출을 활용하여 패킷 동기화 과정을 단순화한 시스템을 구현하였다. 이는, 수신단에서 필요한 동기화 과정에서 송수신 장비의 잡음과 주변 광원 등에 대한 환경 요소를 고려하여 컬러 바코드의 픽셀 값을 분석하여 파일럿 심볼을 검출하기 위한 별도의 알고리즘 설계가 필요한 기존의 시스템과 비교해볼 때, 수신단에서 송수신 환경에 분석을 실시하지 않고 패킷 동기화를 수행할 수 있다.

따라서, YOLO v3를 활용한 CCB-OCC 시스템은 기존의 CCB-OCC와 달리 수신 영상에 따라 별도의 분석과 복잡한 알고리즘 설계 없이 컬러 바코드 영역 추정과 파일럿 심볼 검출을 실시할 수 있으며 기존의 시스템보다 안정적인 성능을 보여준다고 볼 수 있다.

본 시스템에서는 패킷 동기화 단계에서만 YOLOv3를 활용하여 수신 및 디코딩 알고리즘의 복잡도를 낮출 수 있도록 하였으나, 패킷 동기화 단계 이후의 데이터 디코딩 단계에도 딥러닝 알고리즘을 적용한다면, 실행 구조가 보다 단순화된 상태에서도 안정적인 수신 성능을 보장하는 시스템을 구현할 수 있을 것으로 기대된다.

[Reference]

<https://www.hindawi.com/journals/wcmc/2020/3898427/>

<https://arxiv.org/pdf/1506.02640>

<https://arxiv.org/pdf/1804.02767>