

무인기를 이용한 심층 신경망 기반 해파리 분포 인식 시스템

Deep Neural Network-based Jellyfish Distribution Recognition System Using a UAV

구정모¹, 명현[†]

Jungmo Koo¹, Hyun Myung[†]

Abstract: In this paper, we propose a jellyfish distribution recognition and monitoring system using a UAV (unmanned aerial vehicle). The UAV was designed to satisfy the requirements for flight in ocean environment. The target jellyfish, *Aurelia aurita*, is recognized through convolutional neural network and its distribution is calculated. The modified deep neural network architecture has been developed to have reliable recognition accuracy and fast operation speed. Recognition speed is about 400 times faster than GoogLeNet by using a lightweight network architecture. We also introduce the method for selecting candidates to be used as inputs to the proposed network. The recognition accuracy of the jellyfish is improved by removing the probability value of the meaningless class among the probability vectors of the evaluated input image and re-evaluating it by normalization. The jellyfish distribution is calculated based on the unit jellyfish image recognized. The distribution level is defined by using the novelty concept of the distribution map buffer.

Keywords: Convolutional Neural Network, Monitoring System, Unmanned Aerial Vehicle, Jellyfish

1. 서 론

최근 10여 년간 해파리 개체 수는 현격하게 증가하였다. 그 럽에도 불구하고 제대로 된 해파리 개체 관리 시스템이 전무 하다. 국내의 경우 2001년에는 경북 울진 원자력 발전소의 취 수구를 해파리 무리가 막아 발전기 가동이 여러 차례 중단되 는 사고가 있었다. 또한 2012년에는 인천에서 8세 어린이가 해 파리에 쏘여 숨지는 사고도 발생하였다. 해외의 경우 2006년 에 호주의 Brisbane에 정박해 있던 미국 군함의 발전기가 수천 마리의 해파리 무리에 의해 중단되기도 하였다. 또한 2013년 에는 스웨덴의 Oskarshamn 발전소의 냉각수 파이프를 해파리

무리가 막아 가동이 중단되는 사고도 있었다. 어업 손실, 발전 소 가동 장애 등 해파리로 인한 피해가 한국에서만 매년 3,000 억 원 수준으로 심각한 문제로 대두되고 있다.

이러한 문제를 다개체 로봇 시스템으로 해결하고자 연구가 진행되었다¹⁾. 해파리는 몸체의 90% 이상이 수분으로 구성되 어 있기 때문에 불투명한 색을 띄고 모양이 정해져 있지 않다. 따라서 [Fig. 1]에서 보는 것과 같이 육안으로 해파리를 구분하 는 것도 쉽지 않다. 이런 특징을 고전적인 물체 인식 알고리즘 으로는 찾아내기 어렵다.

본 논문에서는 convolutional neural network 구조를 활용하 여 해파리 무리를 찾는 방법을 제안하고자 한다. 또한 해양 환 경에서 신속하게 해파리 분포를 파악하기 위해서는 수면을 한 눈에 볼 수 있어야 한다. 때문에 무인기(unmanned aerial vehicle) 형태의 로봇을 이용하여 시스템을 구성하였다. 감시 시스템에 서 무인기를 활용하면 효율성을 극대화할 수 있다. 무인기는 적은 비용으로 간단하게 항공 이미지를 확보할 수 있으며 상 대적으로 크기가 작아 유지 보수에 용이하다. 또한 조작성이 간 편하며 속도가 빨라 많은 활용성이 있다. 이렇듯 무인기를 이 용한 시스템에 인공지능을 적용함으로써 더욱 정확하고 빠르 게 해파리 분포를 파악하고 위험 상황에 대처할 수 있다. 또한

Received : Sep. 7. 2017; Revised : Oct. 26. 2017; Accepted : Oct. 30. 2017

※ This research was financially supported by the Robot industrial cluster construction program through the Ministry of Trade, Industry & Energy (MOTIE) and Korea Institute for Advancement of Technology (KIAT). This work was also supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (NRF-2015R1A2A1A15056262). The students are supported by the Ministry of Land, Infrastructure and Transport (MoLIT) under the U-City Master and Doctor Course Grant Program

1. KAIST, Daejeon, Korea (jungmokoo@kaist.ac.kr)

† Corresponding author: Civil & Environmental Engineering, KAIST, Daejeon, Korea (hmyung@kaist.ac.kr)



[Fig. 1] Image of jellyfish group on sea surface captured by a unmanned aerial vehicle

JEROS (jellyfish removal robot system)^[1]와의 협업을 통해 더욱 향상된 방식의 해파리 개체 관리 시스템을 운영할 수 있다.

최근에는 무인 비행체가 활성화되면서 항공사진을 활용한 물체 인식 연구도 활발히 진행되고 있다. Li^[2]는 무인 비행체로부터 이미지를 받아 허프 변환(Hough transform)을 적용해 전력선을 검출 및 점검하는 연구를 진행하였다. 또한 Souza^[3]는 항공사진 속의 식물 이미지 정보를 이용하여 다층 신경망 방식으로 병에 걸린 식물을 감지하는 연구를 진행하였다. 이렇듯 무인 비행체는 넓은 영역을 탐사하는 데 효율적이고 물체 인식 알고리즘과 결합하여 많은 분야에 기여를 하고 있다. 본 논문에서도 넓은 해양 환경을 탐사해야하기 때문에 로봇 플랫폼으로써 무인 비행체를 선택하였다.

또한 해파리를 자동 인식하는 연구도 이미 진행된 사례가 있다. Rife^[4]는 원격 조종 무인 잠수정(remotely operated vehicle) 과 자율 무인 잠수정(autonomous underwater vehicle)을 이용해 촬영한 해저 영상에서 해파리를 검출하는 알고리즘을 연구하였다. 이 논문의 알고리즘은 형태학적 변화와 배경의 차이를 이용해 해파리를 추적한다. 위의 연구 모두 해파리 인식에서 전통적인 영상 처리 기법을 사용하고 있다. 이러한 방식은 해파리와 같이 특징이 모호한 물체를 검출하는 데 한계가 있다. 본 논문에서는 이러한 한계를 인공지능 알고리즘을 통해 극복하고자 하였다.

본 논문에서는 인식 정확도 90% 이상과 해파리 분포 인식 알고리즘이 CPU만으로 6.3 Hz 이상 동작하는 것을 목표로 한다. 본 논문에서는 정확한 해파리 개체의 인식이 아닌 전체적인 해파리 분포도 측정을 목적으로 설정하였기 때문에 인식을 90% 수준이면 충분하다고 판단하였다. 또한 무인기는 해상에서 고도 4 m 내외, 속도 2 m/s로 등속 비행하며 설치된 카메라의 시야각(FOV, field of view)를 고려하였을 때 무인기의 진행 방향에서 새롭게 등장한 해파리가 시야에서 사라질 때까지 최소 10번 이상 인식되는 것을 목표로 한다. 이러한 조건을 만족하기 위해서는 최소 6.3 Hz 이상의 인식 속도를 만족해야 한다.

또한 단순히 해파리 개체 수를 세는 것이 아닌 분포도 이미지를 생성하는 새로운 측정 방식을 정의한다.

2. 로봇 시스템

본 장에서는 해파리 분포 모니터링에 사용된 로봇 시스템에 대하여 설명한다. 무인기 형태의 로봇은 감시 체계를 구축하기에 효율적이다. 때문에 다양한 감시 체계에서 무인기를 활용하고 있으며 감시뿐만 아니라 다양한 임무를 수행하고 있다. 또한 다른 감시 시스템에 비해 상대적으로 비용이 적게 들고 필요에 따라 용도의 변화가 가능하며 유지 보수가 용이하다는 것도 장점으로 꼽힌다. 추가적으로 부피가 작고 무게가 가벼워 이동성까지 확보할 수 있다. 본 논문에서 무인기를 활용하는 목적은 항공사진을 바탕으로 한 눈에 해파리 분포를 파악하기 위함이다. 이러한 로봇 시스템을 구축하기 위한 하드웨어에 대해서 소개한다. 또한 해파리 분포 인식 시스템에서 필요한 성능이 어떤 것인지 명시할 것이다.

2.1 하드웨어 구성

해파리 분포 모니터링 시스템을 위한 무인기를 구성하기 위해 필요한 성능을 확인한다. 첫째로, 해양 환경에서 안정적인 비행이 가능해야 한다. 해양 환경에서는 바람이 강하게 불기 때문에 비행체의 자세 제어가 어렵다. 때문에 적당한 하중이 필요하고 관성측정장치(IMU, inertial measurement unit) 등의 센서를 이용한 자세 제어가 가능해야 한다. 기본적인 자세 제어가 완벽하지 않을 경우에는 이착륙 및 비행이 불가능하다. 특히 해양 환경에서는 이러한 부분이 미흡할 경우 큰 사고로 이어질 수 있기 때문에 중요한 조건이다. 둘째, 바다 위에 불시착할 경우를 대비하여 방수가 가능해야 한다. 비행 중 육지까지의 거리가 너무 멀고 배터리의 잔량이 얼마 남지 않았을 때는 바다 위에 착륙해야 한다. 이러한 경우에 방수가 되지 않는다면 모든 전자 장비들은 제대로 작동하지 못할 것이다. 이뿐만 아니라 비행 중 비가 오는 상황에도 대비해야 한다. 셋째, 15분 이상의 비행시간을 확보해야 한다. 해양 환경과 같은 넓은 지역을 감시하기 위해서는 충분한 비행시간이 필요하다. 해파리 분포 파악을 위한 일정한 속도로 넓은 지역을 스캔하고 다시 돌아오기 위해 약 15분 이상의 비행시간은 필수적이다.

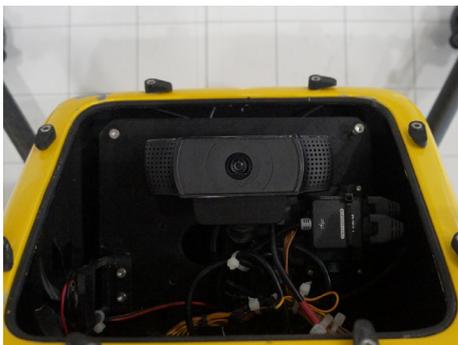
사용하는 무인기 모델인 HexH2O [Fig. 2]는 DJI 사의 NAZA V2 제어기를 이용한 본체에 유리섬유(FRP, fiber reinforced plastics)를 이용하여 방수 작업을 마친 형태이다. 총 무게는 1.4 kg이며 크기는 폭 740 × 높이 240 × 길이 650 mm를 가지고 있다. 배터리는 2개까지 장착 가능하며 총 하중은 2 kg이다. 또

한 LiPo 6S 1400 mAh의 배터리 용량으로 25분까지 비행이 가능하며 6개의 프로펠러를 이용하기 때문에 25 mph의 바람에도 안정적으로 비행할 수 있다. 비행 중 호버링 기능에 대한 정확도는 GPS 모드에서 수직으로 0.8 m 이고 수평으로 2.5 m이다. 또한 조종기는 2.4 Hz로 통신하며 오픈된 공간에서 통신 가능 거리는 1,000 m이다.

이와 같은 무인기가 본 논문에서 목적으로 하는 기능을 수행하기 위해서는 연산장치가 필요하다. 무게가 가벼우면서 좋은 성능을 내기 위한 연산장치를 모색하였고 인텔 NUC i7 모델을 선정하였다. 모델명은 BOXNUC5i7RYH으로 5세대 인텔 코어 i7-5557U 프로세서가 장착되어 있다. 본래 HexH2O에는 고프로(GoPro) 카메라를 장착할 수 있는 짐벌이 설치되어 있다. 하지만 고프로 카메라는 컴퓨터로 실시간 영상 전송이 불가능하기 때문에 기존에 장착되어 있던 짐벌을 제거하고 그곳에 Logitech C920 webcam을 설치하였다. 무인기는 비행시 진동이 있기 때문에 실리콘 재질로 되어 있는 충격 완화 장치와 함께 [Fig. 3]과 같이 무인기 앞 쪽에 카메라를 설치하였다. 해양 환경에서 사용하기에 필수적인 자동 조도 보정 기능으로 햇빛이 강한 날이나 흐린 날에도 좋은 품질의 영상을 받을 수



[Fig. 2] Waterproof type drone HexH2O. It is manufactured by QuadH2O company and uses handmade frame. The maximum speed is 56 km/h

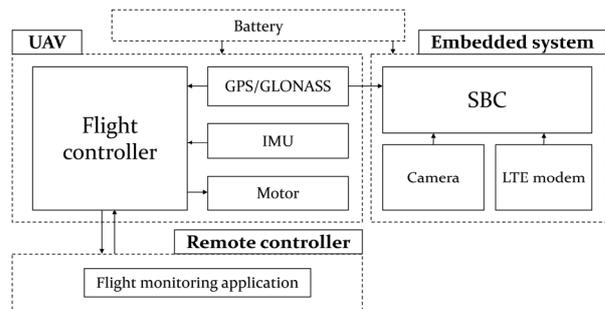


[Fig. 3] C920 mounted on HexH2O

있다. 영상의 최대 프레임 속도는 30 fps이다. 해파리 분포에 대한 원격 모니터링을 진행하기 위해서는 원활한 통신이 필요하다. 해상에서 블루투스나 와이파이와 같은 통신으로는 한계가 존재하고 인터넷에 연결하여 구글 맵을 전송받아야 하기 때문에 충분한 통신 속도가 보장되어야 한다. 따라서 4G LTE (long term evolution) 통신을 사용하였다. GPS 좌표에 대한 정보를 컴퓨터로 전송받기 위해 NAZA GPS 신호를 UART (universal asynchronous receiver/transmitter) 통신으로 받았다. USB to UART 모듈을 이용하여 컴퓨터의 USB에 연결하였으며 패킷을 복호화하여 GPS 좌표를 받아왔다. 해당 위치 인식 모듈은 GPS뿐만 아니라 GLONASS도 지원하여 더욱 정확한 위치를 알 수 있다. 또한 기압계 센서도 융합되어 더욱 정확한 높이 정보도 함께 받아볼 수 있다.

2.2 기능

본 절에서는 전체적인 하드웨어 구성과 각각의 연결 관계에 대하여 설명한다. [Fig. 4]에서 보이는 것처럼 크게 무인기 부분과 임베디드 시스템 부분, 원격 제어기 부분으로 나눌 수 있다. 첫째로, 무인기 부분에서는 기본적인 비행 제어를 담당한다. NAZA V2 비행 제어 장치에 GPS 모듈과 IMU, 모터 등이 연결되어 있다. 다양한 센서와 구동부가 연결되어 비행 자세 제어, 속도 제어, 위치 인식을 수행하며 비행을 원활하게 수행할 수 있도록 도와준다. 둘째로, 임베디드 시스템에서는 단일보드컴퓨터(SBC, single board computer)를 중심으로 카메라와 LTE 통신 모듈, GPS 모듈을 연결한다. 해파리 분포 인식 연산을 위한 컴퓨터가 필요하며 인식하기 위한 영상을 받을 수 있도록 카메라가 연결되어 있다. 또한 서버에서 해파리 분포에 대한 모니터링을 수행하기 위해 LTE 통신으로 정보를 전송한다. 셋째로, 무인기의 원격 제어기로 조종이 가능한데 비행을 위한 준비 과정 또는 조종을 하는 동안의 모든 과정을 스마트폰 앱을 통해 확인할 수 있다.



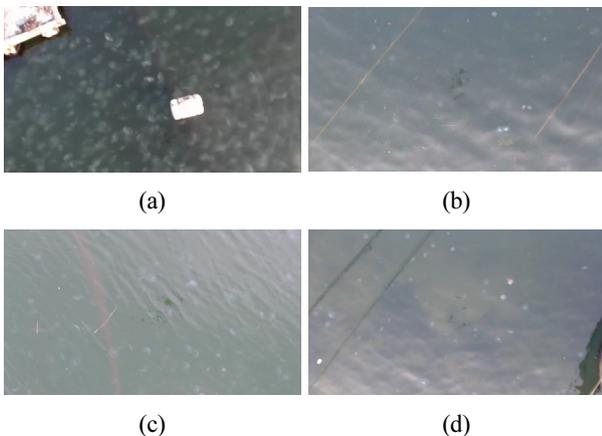
[Fig. 4] Overall robot hardware configuration and function diagram

3. 심층 신경망

본 장에서는 해파리 분포 인식을 위한 심층 신경망 개발 과정에 대하여 소개한다. 본 논문에서 목적으로 하는 로봇 플랫폼을 사용한 실시간 해파리 분포 인식 실현을 위해 높은 인식률을 가지면서 연산 속도가 빠른 신경망 구조를 개발하는데 초점을 맞추었다. 이를 위해 LeNet-5^[5], AlexNet^[6], GoogLeNet^[7]과 같은 잘 알려진 신경망 구조를 참고로 하였으며 실험적인 과정으로 다양한 구조를 만들어 평가해 봄으로써 가장 적합한 구조를 도출하였다. 하위 절에서 설명할 내용으로는 해파리 분포 인식을 위한 데이터 셋을 만든 과정과 그 데이터 셋의 특징에 대해서 살펴본다. 10가지로 구분한 해양 환경 데이터의 수학적 분포를 살펴보고 그 상관관계에 대해서 분석한다. 또한 제한하는 신경망 구조를 학습시키는 과정과 학습의 하이퍼 파라미터(hyperparameter) 설정에 대해서 알아보고 그에 대한 학습 결과를 보여줄 것이다.

3.1 학습 데이터

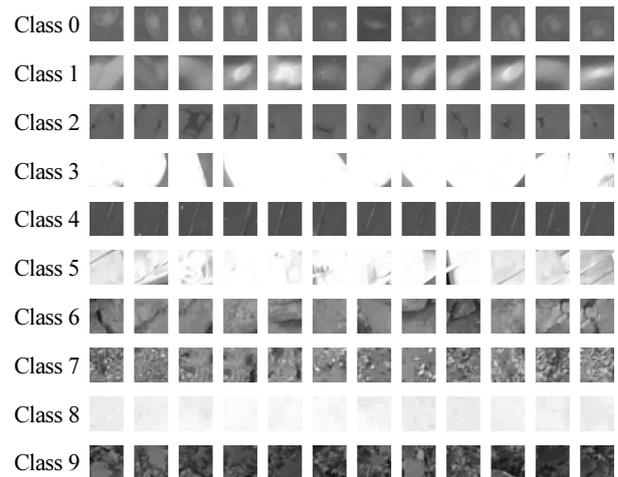
신경망을 학습시키기 위해서는 목적에 맞는 학습 데이터가 필요하다. 학습 데이터는 마산과 고성 등에서 약 2년에 걸쳐 수집되었다. 항상 다른 날씨와 시간에 데이터를 수집하였기 때문에 탁도 및 파고도 다양하게 적용되어있다. 하지만 무인기가 비행가능한 상황만 고려했기 때문에 비가 많이 내리거나 바람이 너무 거세게 부는 경우는 제외되었다. 무인 비행체를 바다 위로 비행시켜 영상을 녹화하였다. 이때 비행의 고도는 해파리 분포를 파악할 수 있다고 판단되는 약 4~5 m로 제한하였다. [Fig. 5]와 같이 다양하게 수집된 영상에서 해파리와 그



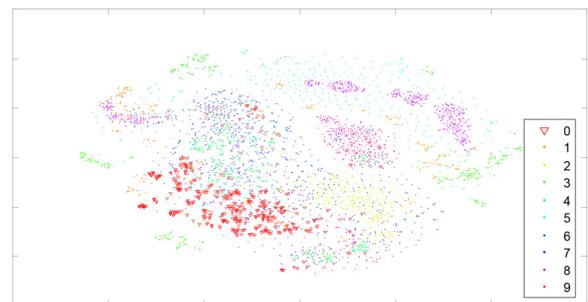
[Fig. 5] Examples of images acquired using unmanned aerial vehicles. (a) Masan, July 15, 2015. (b) Goseong, Aug. 13, 2015. (c) Goseong, July 18, 2016. (d) Masan, Sep. 12, 2016

외의 해양 환경 물체들을 10가지로 구분하여 32×32 픽셀 크기로 잘라내었다. 또한 수면이나 부표, 밧줄과 같은 물체들에 대해서도 같은 방식으로 이미지를 생성하였다. 그렇게 만들어진 학습 데이터는 약 1만장이다. [Fig. 6]에 보이는 것과 같이 클래스 0번은 해파리(보름달물 해파리, *Aurelia aurita*), 1번은 해수면, 2번은 음영, 3번은 부표, 4번은 밧줄, 5번은 선박, 6번은 암석, 7번은 자갈, 8번은 땅 그리고 9번은 풀이다.

수집된 학습 데이터의 관계를 확인하기 위해 t-Distributed Stochastic Neighbor Embedding (t-SNE)^[8]를 이용하여 시각화하였다. t-SNE는 고차원의 데이터들의 차원을 감소시키며 비슷한 정보를 가진 데이터들을 묶어 데이터의 분포를 확인할 수 있게 해준다. Matlab을 이용하여 학습 데이터를 모두 t-SNE 가시화 알고리즘을 수행하기 위한 형태로 변환한 후 차원(dimension)은 2, 복잡도(perplexity)는 10으로 설정하여 알고리즘을 적용하였다. 알고리즘을 수행한 결과는 [Fig. 7]에 보이



[Fig. 6] Examples of cropped images for each class. Class 0 is the jellyfish, class 1 is the sea surface, class 2 is the dark area, class 3 is the buoy, class 4 is the rope, class 5 is the ship, class 6 is the rock, class 7 is the gravel, class 8 is the ground, and class 9 is the grass



[Fig. 7] Distribution of training data using t-SNE visualization

는 것과 같은 분포를 띄고 있다. 삼각형 모양의 데이터가 해파리 학습 데이터를 의미한다. 각 클래스 간 데이터에 상관관계가 크게 나타났지만 충분히 구분될 수 있는 수준의 분포를 나타내고 있다.

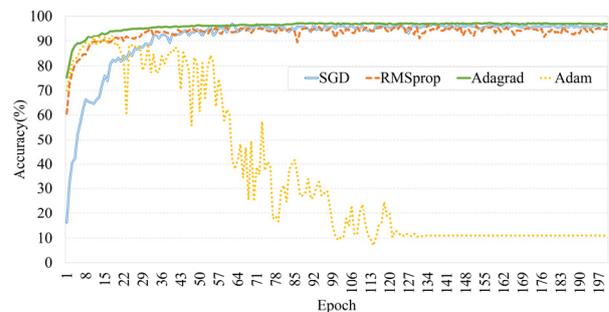
3.2 신경망 구조

본 절에서는 제안하는 신경망 구조에 대해 기술한다. 제안하는 신경망 구조는 다양한 신경망 구조를 참고하고 이를 변형하여 만든 후 평가하는 방식으로 만들어졌다. 즉, 실험적으로 평가하여 높은 성능을 가지는 신경망 구조를 찾아내었다. 본 연구에서 목표로 하는 신경망 구조는 인식 정확도와 인식 속도 사이의 적절한 균형에 초점을 맞추어 개발되었다. 이는 로봇 플랫폼에서 저전력으로 연산해야 하는 조건에 충족하기 위함이다. 다양하게 만들어진 신경망 구조는 모두 같은 학습 데이터로 학습되었다. 그 결과를 확인하여 인식 정확도와 연산 속도 면에서 가장 효율적인 구조가 선택되었다. 제안하는 신경망 구조는 [Table 1]과 같은 형태로 구성되어 있다. 이러한 구조를 선택하게 된 이유로 가장 먼저 연산 속도를 꼽을 수 있다. 평가를 진행한 신경망 구조들 중 가장 빠른 연산 속도를 보인 것은 LeNet-5이다. 하지만 이 신경망 구조를 해파리 분포 인식 알고리즘에 적용할 시 평균 3.01 Hz의 연산속도를 보여서 본 논문에서 목표로 하는 연산속도인 6.3 Hz 이상을 가지지 못한다. 그러므로 신경망 연산 속도를 대폭 개선해야 하며 동시에 인식을 하락을 최소화해야 한다. 이러한 목적으로 연산의 병목 현상이 가장 많이 발생하는 fully connected layer를 줄여야 한다. 하지만 동일한 구조에서 fully connected layer만 강제로 줄이게 되면 인식이 대폭 하락할 수 있으므로 신경망 구조에서 선행되는 컨볼루션 레이어의 특징 추출 능력을 강조시켜 줘야 한다. 즉, 제안하는 신경망 구조의 전체적인 개념은 마지막단의 회귀 분석(regression)을 최소화하는 대신 특징 추출을 강화하여 인식을 하락을 최소화하는 것이다.

3.3 신경망 학습

본 절에서는 제안하는 신경망 구조를 학습시키는 과정에 대해서 소개한다. 위에서 소개한 신경망 구조와 더불어 해양 환경 데이터를 적용하여 학습을 진행하면서 설정한 하이퍼파라미터에 대한 분석도 언급한다. 학습을 진행한 기본적인 조건은 다음과 같다. 미니 배치의 크기는 2로, epoch는 200번으로 설정하여 학습을 진행했다. 손실 함수는 평균 자승 오차(mean square error)를 사용했고 학습의 정확도를 위해 다양한 최적화 방법을 적용하여 적합한 방식을 찾아내었다.

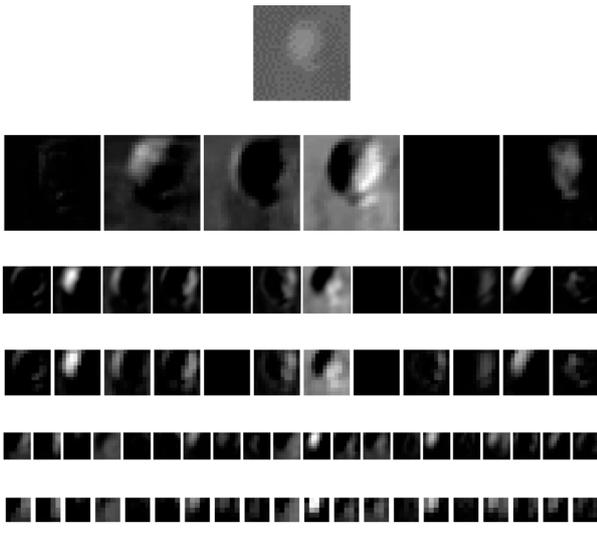
제안하는 신경망을 학습시키기 위하여 4가지 최적화기(optimizer)를 적용하고 인식률의 수렴 정도를 비교해보았다. [Fig. 8]에서 보는 것과 같이 적용한 최적화 중 Adagrad^[9]가 제안한 신경망 구조에서 높은 정확도로 수렴하는 것을 확인할 수 있다. 제안한 신경망 구조는 층이 많지 않기 때문에 Adam^[10]과 같이 빠른 수렴 속도를 가지는 optimizer보다 Adagrad가 더욱 적합하다. Adagrad는 빈번한 데이터에 대해서 업데이트를 적게 수행하고, 빈번하지 않은 데이터는 업데이트를 크게 수행하므로 넓게 퍼진 데이터에 적합하다. 본 논문에서 다루는 데이터는 군집이 명확히 생성되지 않은 퍼져 있는 형태를 띄고 있으므로 해당 optimizer가 가장 적합하게 작동한다. 최종 학습된 kernel에 의해 생성된 각 층에 대한 특징 맵의 예시는 [Fig. 9]와 같다.



[Fig. 8] Comparison of training accuracy convergence result depending on the optimizer

[Table 1] Proposed network architecture and parameters setting

Layer	Input channels & size	Filter size	Stride	Type	Output channels & size
Convolution	1 (32 × 32)	5 × 5	1	Leaky ReLU	6 (28 × 28)
Convolution	6 (28 × 28)	5 × 5	1	Leaky ReLU	12 (24 × 24)
Pooling	12 (24 × 24)	2 × 2	2	Max	12 (12 × 12)
Convolution	12 (12 × 12)	3 × 3	1	Leaky ReLU	20 (10 × 10)
Pooling	20 (10 × 10)	2 × 2	2	Max	20 (5 × 5)
Convolution	20 (5 × 5)	5 × 5	1	Leaky ReLU	160 (1 × 1)
Fully connected	160 (1 × 1)	-	-	Softmax	10 (1 × 1)



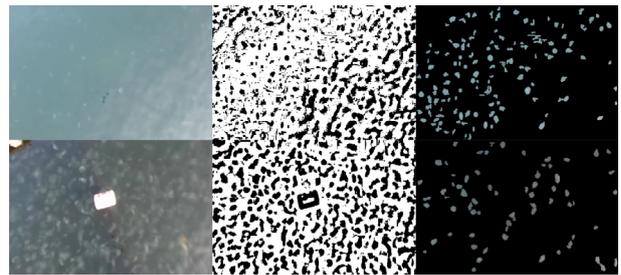
[Fig. 9] Visualization of feature maps for a jellyfish image

4. 해파리 분포 인식 알고리즘

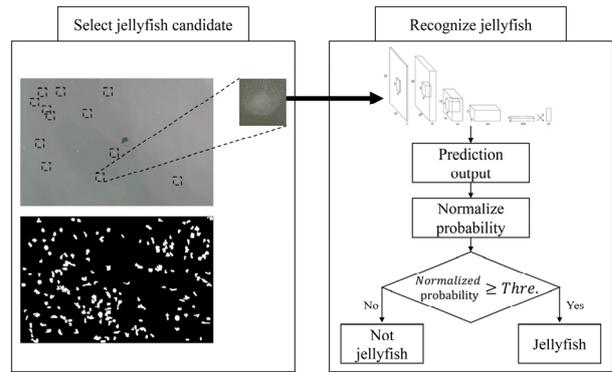
본 장에서는 해파리 분포 인식을 위한 과정을 소개한다. 본 논문에서 제안하는 신경망을 바탕으로 알고리즘을 구성하였으며 해당 알고리즘이 해파리 분포 인식에 있어서 좋은 성능을 발휘할 수 있게 하는 전처리 및 후처리 과정에 대해 소개한다. 학습된 신경망을 이용하여 실시간 성능을 발휘하기 위해서는 한 프레임 안에서 입력 영상으로 사용되는 후보군의 개수가 중요하다. 그 개수를 줄일수록 프레임 처리 속도는 빨라진다. 하지만 후보군을 너무 적게 잡으면 해파리를 놓칠 수 있으므로 명확히 해파리가 아닌 부분만을 제거해야 한다. 이러한 기능을 수행하기 위해 *locally adaptive threshold*를 사용했고 그 결과를 바탕으로 컨투어를 찾아 노이즈 및 해파리의 크기가 아닌 물체들을 모두 제거한다. 이렇게 완성된 후보군을 모두 신경망의 입력으로 하여 예측하고 그 확률을 바탕으로 해파리 분포 정도를 계산한다. 해파리 분포도를 계산할 때 시간의 흐름을 고려한 분포도 맵을 생성하는 방법에 대해서도 소개한다.

4.1 해파리 후보군 선택

위에서 사전 학습된 신경망을 기반으로 해파리 분포 인식 알고리즘은 설계되었다. 무인 비행체가 수면 위를 비행하며 촬영하는 영상에서 해파리를 찾아내기 위해서는 전처리 과정과 후처리 과정이 필요하다. 첫 번째로 원본 영상이 들어오면 *locally adaptive threshold*를 통해 배경을 제거한다. *Locally adaptive threshold*는 해당 픽셀 주변 영역 값의 평균을 임계치



[Fig. 10] Locally adaptive threshold example. (Left) Original image. (Center) Result image using locally adaptive threshold. (Right) Result image through contour filter



[Fig. 11] Flowchart of jellyfish recognition algorithm

로 사용하는 방법이다.

이렇게 얻은 결과 영상에서 노이즈라고 판단할 수 있는 영역과 해파리의 크기가 아닌 물체는 모두 제거된다. 제거하기 위해 컨투어 그룹을 생성하고 각 컨투어의 크기를 기준으로 필터링이 진행된다. 이러한 전처리 과정을 통해 [Fig. 10]에 보이는 것과 같은 결과를 얻을 수 있다.

이러한 전처리 과정은 신경망의 계산량을 줄여 줄 수 있어 실시간 성능에 직접적인 영향을 미친다. 필수적인 후보들만 골라낼수록 인식 속도는 증가한다. 하나의 프레임을 처리하기 위해 신경망에서 *forward pass*를 여러 번 할수록 그 수만큼 인식 시간은 비례하여 증가한다. 전처리 과정이 끝난 영상에 남아있는 컨투어 중심점을 기준으로 32×32 크기의 이미지를 잘라낸다. 잘라낸 이미지는 모두 신경망의 입력 이미지가 된다. [Fig. 11]에서 보는 것과 같이 신경망을 통해 모든 입력 이미지를 *forward pass*하게 되면 각 입력 후보에 대한 해파리 판별 결과가 도출된다.

4.2 해파리 분포도

신경망을 거친 후에 10개의 클래스에 해당하는 예측 값 P 가 생성된다.

$$P = \{P_{class0}, P_{class1}, P_{class2}, \dots, P_{class9}\} \quad (1)$$

이 중 충분히 낮은 확률을 가진 클래스를 모두 제거한다. 제거할 때는 적당한 임계치를 적용하여 제거하는데 본 논문에서는 0.1의 값으로 설정하였다.

$$Threshold(P) = \begin{cases} P_{class i}, & \text{if } P_{class i} > threshold \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

이 과정을 거쳐 만들어진 P' 은 다음과 같다.

$$P' = \{P'_{class0}, P'_{class1}, P'_{class2}, \dots, P'_{class9}\} \quad (3)$$

이를 아래와 같이 정규화시키면

$$R = \frac{1}{\sum_i P'_{class i}} \cdot P' \quad (4)$$

결과적으로 아래와 같은 정규화된 확률 벡터 R 을 얻을 수 있다.

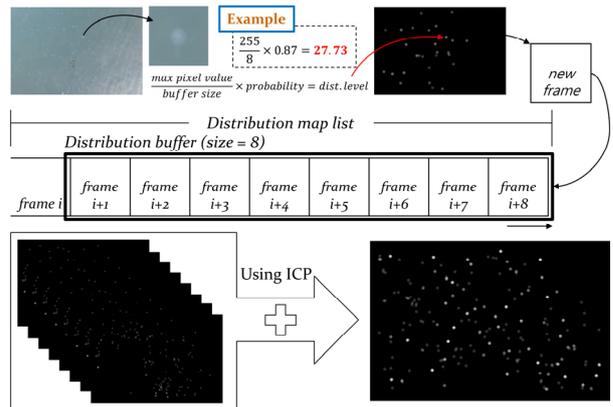
$$R = \{R_{class0}, R_{class1}, R_{class2}, \dots, R_{class9}\} \quad (5)$$

본 논문에서는 0번 클래스를 해파리로 설정했으며 0번 클래스에 대한 확률 임계치 값은 0.8로 설정했다. 그래서 R_{class0} 이 0.8보다 크면 해당 입력 이미지에 해파리가 있다고 판단한다. 이러한 과정을 반복하여 한 프레임에서 추려진 후보군에 대한 해파리를 모두 인식한다. 이 과정에서 해파리 분포 맵을 만들는데 먼저 분포 맵 리스트를 만들고 그 리스트의 버퍼 크기를 설정한다. 본 논문에서는 버퍼 크기를 8로 설정했다. 이는 8 프레임을 모두 조합하여 해파리 분포를 파악한다는 의미인데 시간의 흐름을 반영하여 강인한 해파리 분포 인식을 달성하기 위함이다. 경우에 따라 수면 위의 거품 또는 햇빛 반사와 같은 조건 때문에 해파리라고 오인식이 되는 경우가 있는데 여러 프레임을 조합하여 판단하게 되면 시간이 지남에 따라 오인식을 줄일 수 있기 때문에 이와 같은 방식을 적용한다. 버퍼에 쌓인 이미지를 모두 더할 때는 시간이 흐름에 따라 해파리의 위치가 변하기 때문에 ICP (iterative closest point)^[11]를 이용하여 영상의 회전과 병진을 고려하여 합산한다. 한 프레임에서 ICP 알고리즘을 처리하는데 소요되는 시간은 최대 24 ms, 최소 2 ms로 매우 빠르게 수행되므로 전체 알고리즘의 실시간성을 해치지 않는 정도의 수준이다. 분포 맵을 만들 때는 우선 원본 프레임과 같은 크기의 이미지를 생성하고 모든 픽셀 값을 0으로 설정한다. 그 후 해파리가 인식될 때 마다 인식

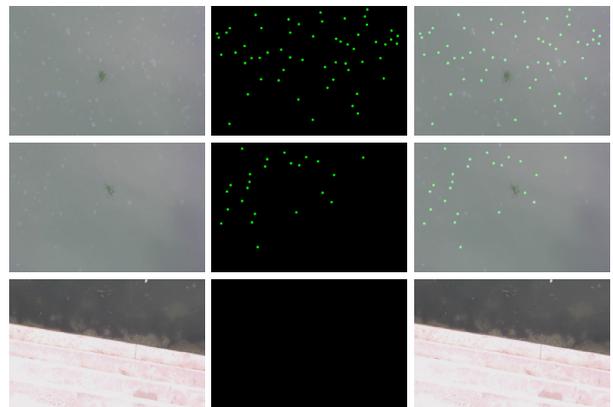
된 위치에 계산된 분포 레벨(distribution level)을 저장한다. 식 (6)과 같은 계산 과정을 거치는데 R_{class0} 는 해파리일 확률이다. 채워진 모든 버퍼의 R_{class0} 이 모두 1이면 식 (6)과 같은 계산을 거친 분포 레벨을 모두 더했을 때 픽셀의 최댓값(255)이 될 것이고 그것을 영상화시키면 가장 밝은 값을 가지게 될 것이다. 어두울수록 해당 위치에 해파리가 존재할 확률이 적고 밝을수록 해당 위치에 해파리가 분포해 있을 확률은 높다.

$$\text{Distribution level} = \frac{\text{Max. pixel value}}{\text{buffer size}} \times R_{class0} \quad (6)$$

이러한 과정을 인식된 모든 해파리에 적용하면 그 곳에 분포해 있을 확률을 한 장의 이미지를 통해 가시적으로 확인할 수 있으며 또한 [Fig. 12]에서 보는 것과 같이 마지막에 생성된 이미지는 각 위치마다 해파리가 분포해 있을 정도를 의미하



[Fig. 12] Example of generating a distribution map and saving it in map list. Unit distribution maps are combined to make a final distribution map. In this example, maximum pixel value is 255 and buffer size is 8



[Fig. 13] Examples of result images. (Middle) Original images. (Right) Result images of jellyfish distribution recognition algorithm. (Right) Overlapping image

[Table 2] Comparison of recognition accuracy and recognition speed by networks

	Proposed	LeNet-5	AlexNet	GoogLeNet
Accuracy (%)	94.06	93.57	95.60	98.37
Frame rate (Hz)	8.23	3.01	0.01	0.02

로 모든 픽셀 값을 합하여 현재 로봇이 위치한 지역의 해파리 분포도를 표현할 수 있다. [Fig. 13]에서 보는 것과 같이 원본 영상에 대하여 분포도 이미지를 표현할 수 있으며 해파리 분포가 없는 경우와 존재하는 경우의 차이를 분포도 이미지를 통해 가지적으로 한 눈에 확인할 수 있다.

5. 실험

본 논문에서 제안한 알고리즘을 평가하기 위해 다양한 신경망 구조에 해양 환경 데이터를 이용하여 학습을 진행했다. 학습을 진행한 신경망 구조는 널리 알려진 LeNet-5, AlexNet, GoogLeNet이다. 모든 신경망은 같은 조건에서 학습하였고 학습은 NVIDIA DIGITS^[12]를 이용하여 진행했다. 학습된 신경망 구조를 본 논문에서 제안하는 해파리 분포 인식 알고리즘에 적용하여 성능을 검증하였다.

[Table 2]에서 보는 것과 같이 가장 좋은 인식률을 보인 신경망 구조는 GoogLeNet이다. 제안한 신경망은 GoogLeNet 대비 인식률은 4.31%p 낮지만 인식 속도에서는 400배 빠르다. 또한 AlexNet 대비 인식률은 1.54%p 만큼 낮지만 인식 속도에서는 800배 이상 빠른 성능을 보이고 있다. LeNet-5 대비 인식률은 1.05%p 만큼 낮지만 인식 속도에서는 약 1.68배 빠르다. 결론적으로 제안한 신경망 구조는 단일보드컴퓨터에서 CPU만으로 연산을 수행할 때 가장 적합한 구조를 가지고 있다고 할 수 있다. 결론적으로 무인기 내의 단일보드컴퓨터에서 신속한 해파리 분포 인식을 수행할 수 있는 성능을 입증하였다.

6. 결론

전지구적 문제인 해파리 개체 증가를 로봇과 인공지능을 결합하여 해결하고자 하였다. 해파리의 특징을 정의하는 것은 매우 어려운 일이다. 때문에 해파리를 특징 기반의 영상 인식 알고리즘으로 찾아내는 것은 한계가 있다. 그러므로 본 논문에서는 인공지능을 활용한 접근법을 시도했다. 결과적으로 해파리 인식에서 좋은 성능을 보였고 해파리 단일 이미지를 기반으로 한 분포 인식에도 훌륭한 성능을 보여주었다. 또한 해파리 분포 인식에 가장 적합한 로봇의 형태를 선택하였고 해양 환경에서 적절한 작동이 가능하게 하였다. 제안한 신경망 구조는 제한된

연산 장치에서 8 Hz 이상의 인식 속도를 달성하여 로봇을 이용한 실시간 모니터링 시스템을 가능하게 하였다. 후에 무인기의 자율주행을 추가하여 원하는 지역을 손쉽게 정찰할 수 있도록 개선할 것이며 인식할 수 있는 해파리를 보름달물해파리에 그치지 않고 다양한 종류를 인식할 수 있게 개선해야 할 것이다.

References

- [1] D. Kim, J.-U. Shin, H. Kim, H. Kim, D. Lee, S.-M. Lee, and H. Myung, "Design and implementation of unmanned surface vehicle JEROS for jellyfish removal," *Journal of Korea Robotics Society*, vol. 8, no. 1, pp. 51-57, 2013.
- [2] Z. Li, Y. Liu, R. Walker, R. Hayward, and J. Zhang, "Towards automatic power line detection for a UAV surveillance system using pulse coupled neural filter and an improved Hough transform," *Machine Vision and Applications*, vol. 21, no. 5, pp. 677-686, 2010.
- [3] J. R. Souza, C. C. Mendes, V. Guizilini, K. C. Vivaldini, A. Colturato, F. Ramos, and D. F. Wolf, "Automatic detection of Ceratocystis wilt in Eucalyptus crops from aerial images," *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3443-3448, 2015.
- [4] J. Rife, and S. M. Rock, "Segmentation methods for visual tracking of deep-ocean jellyfish using a conventional camera," *IEEE Journal of Oceanic Engineering*, vol. 28, no. 4, pp. 595-608, 2003.
- [5] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278-2324, 1998.
- [6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, pp. 1097-1105, 2012.
- [7] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-9, 2015.
- [8] L. V. D. Maaten, and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579-2605, 2008.
- [9] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2121-2159, 2011.
- [10] D. Kingma, and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.
- [11] Z. Zhang, "Iterative point matching for registration of free-form curves and surfaces," *International Journal of Computer Vision*, vol. 13, no. 2, pp. 119-152, 1994.
- [12] NVIDIA, "The NVIDIA Deep Learning GPU Training System (DIGITS)," [Online], <https://developer.nvidia.com/digits>, Accessed: October 30, 2017.



구 정 모

2015 경희대학교 전자전파공학 학사
2017 KAIST 건설 및 환경공학 석사
2017~현재 KAIST 건설 및 환경공학과 박사
과정

관심분야: 영상인식, 기계학습



명 현

1992 KAIST 전기 및 전자공학 학사
1994 KAIST 전기 및 전자공학 석사
1998 KAIST 전기 및 전자공학 박사
1998~2002 한국전자통신연구원(ETRI) 선임
연구원
2002~2003 (주)이머시스 연구소장/기술이사
2003~2008 삼성전자종합기술원 전문연구원
2008~현재 KAIST 건설 및 환경 공학과 교수

관심분야: 로봇 네비게이션, 건설 로봇, 수중/수상 로봇, 인공지능,
기계학습