

드론 활용 목표물 추적 응용에서의 인공지능 작업 실행 효율 비교 분석: 클라우드 기반 대 드론 기반

손경환*, David Hostallero*, 김대우*, 방지환*, 강완주*,
윤세은*, 구윤표*, 여현호*, 하재형*, 한동수*, 이영*

On the Efficiency of Running Machine Learning Tasks for Drone-Based Target Tracking: Cloud-Based vs. Drone-Based

Kyunghwan Son*, David Hostallero*, Daewoo Kim*, Jihwan Bang*, Wan Ju Kang*,
Se-eun Yoon*, Yoon-pyo Koo*, Hyun-ho Yeo*, Jae-hyung Ha*, Dongsu Han*, Yung Yi*

요약

무인항공기(Unmanned Aerial Vehicle 또는 드론)에서 목표물 추적을 위한 2가지 대표적인 인공지능 작업은 (i) 목표물 인식 및 추적과 (ii) 드론의 제어이다. 이와 같은 인공지능 작업들은 높은 실시간 계산을 요구하나 중소형 무인항공기들에서 탑재 가능한 컴퓨팅 능력 한계로 인한 한계가 존재한다. 이를 해결하기 위해 클라우드로 인공지능 작업을 오프로딩하는 방법이 존재하나, 이는 지연시간을 동반하게 되는 단점이 있다. 본 논문에서는 컴퓨팅 능력의 한계가 존재하는 순수 드론위에서의 인공지능 작업 수행과, 지연시간이 존재하는 순수 클라우드 기반의 인공지능 작업 수행의 성능에 대해 무인항공기 기반 목표물 추적 응용에서 비교 분석을 수행한다. 본 결과를 통하여 현재 기술로 달성할 수 있는 정량화 된 무인항공기 기반 목표물 추적 성능과 앞으로 출현하게 될 무인항공기 기반 인공지능 서비스들의 원활한 수행을 위해서 컴퓨팅능력, 통신망 성능들의 요구사항을 확보할 수 있다.

Key Words : drone, target tracking, machine learning, cloud

ABSTRACT

Unmanned Aerial Vehicles, otherwise known as drones, necessitate the execution of two artificial intelligence tasks while tracking arbitrary objects: (i) the detection and tracking of the target visual and (ii) the corresponding control of the drone movement. These tasks are known to require substantial computing power; however, due to the physical limitations of small-sized drones in their accessibility to computing hardware, it is often practically impossible to execute the said tasks in a satisfactory manner. Cloud offloading could alleviate such a problem, but it incurs a cost in the form of network delay. This paper thus presents a comparative analysis between the resource-poor drone setting and the resource-rich, network-delayed cloud setting in their accomplishment of an object tracking mission. Through this analysis, we quantify the drone-based system's tracking performance achievable with existing technology, and also derive the anticipated requirements for computing resources and communication networks for a better accommodation of AI-related services based on drones.

※ 이 논문은 2017년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (No.B0717-17-0034, 서비스/단말/네트워크 다양성 지원을 위한 미래형 다차원 네트워크 시스템 아키텍처 연구)

• First Author : School of Electrical Engineering, KAIST, {kevinson9473, ddhostallero}@kaist.ac.kr, 학생회원

◦ Corresponding Author : School of Electrical Engineering, KAIST, yiyung@kaist.edu, 종신회원

* School of Electrical Engineering, KAIST, {daewookim, jhbang1422, soarhigh, granelle, ypkoo, pkpk5958, thundershower, dongsu_han}@kaist.ac.kr

논문번호 : KICS2017-10-318, Received October 20, 2017; Revised January 8, 2018; Accepted January 15, 2018

1. 서 론

무인항공기(이하 드론)의 개발과 보급이 가속화됨에 따라서, 드론을 활용한 다양한 서비스들이 등장하고 있다. 이와 같은 서비스 및 응용들 중 실시간 인공지능 작업을 요구하는 것들이 증가하고 있으며, 그 중 대표적인 예가 실시간 목표물 추적 서비스이다. 이 서비스는 감시, 정찰, 재난 구조, 야생동물보호 등 여러 가지 목적으로 사용될 수 있는 드론 기반 기본 응용중의 하나이다^[1]. 드론 활용용 목표물 추적은 [그림 1]과 같은 절차를 통해서 이루어진다.

드론 상에서 수행하는 작업은 크게 이미지 처리 모듈, 드론 추적 알고리즘 모듈, 드론 조종 모듈, 네트워킹 모듈로 나뉘며 각 모듈의 역할은 다음과 같다.

(가) 이미지 처리 모듈: 드론에 장착되어 있는 카메라 센서를 통해 받은 영상을 실시간으로 처리한다. 드론에서 인공지능 작업을 하는 경우 사진 상에서 목표물을 인식하여 드론 추적 알고리즘 모듈로 보내주며, 클라우드를 활용하는 경우 클라우드로 영상을 전송한다.

(나) 드론 추적 알고리즘 모듈: 메시지 형태로 전달받은 물체의 위치 및 크기 정보를 활용하여 목표물을 추적하기 위한 드론의 움직임을 결정한다. 결정된 움직임을 드론 조종 모듈로 전송한다.

(다) 드론 조종 모듈: 추적 알고리즘 모듈 혹은 클라우드로부터 받은 드론의 움직임을 실제로 수행한다. 실제로 드론에게 명령을 전달하여 움직이고, 또한 드론으로부터 정보를 받아 GCS 및 드론 추적 알고리즘으로 전달하는 작업을 수행한다.

(라) 네트워킹 모듈: 드론과 GCS 통신을 담당하며 실시간 영상 스트리밍, 상태 보고 등을 수행한다.

위의 절차 중에서 인공지능 작업을 필요로 하는 요소는 다음과 같은 두 요소이다.

- 물체인식 및 추적: 드론의 카메라 센서로부터 받은 영상에서 목표물을 찾아내고 화면상에서의 물체 이동을 지속적으로 추적하는 작업이다. 일반적으로 깊은 신경망 (Deep Neural Network) 구조 기반의 인공지능 알고리즘을 이용하여 목표물의 위치 및 사이즈를 찾아낸다.
- 드론 제어: 한번 인식된 목표물을 지속적으로 카메라 화면 내에 위치시키면서 적절한 거리를 유지하도록 드론의 움직임을 제어하는 작업이다. 정밀도 높은 자율 제어를 위해서 깊은 강화 학습 (Deep Reinforcement Learning)에 기반을 둔 드론 제어 알고리즘을 사용한다.

최근 급속도로 발전하는 프로세서 기술 (CPU 및 GPU)에도 불구하고, 중소형 드론에 탑재되는 컴퓨터는 컴퓨팅 용량의 측면에서 한계를 가질 수밖에 없고, 이와 같은 컴퓨팅 용량으로 위에 언급한 두 가지 인공지능 작업을 완벽히 수행하는 것은 불가능한 실정이다. 특히, 드론에서 목표물 추적을 위해서 해야 하는 기타 작업(예: 목표물 비디오 스트리밍 및 주기적 상태보고)이 존재함으로 인하여, 발생하는 각 작업 간 자원 경쟁이 심한 상황이 이와 같은 인공지능 작업 수행을 위한 컴퓨팅 능력의 확보를 어렵게 한다. 대안으로 생각할 수 있는 방법은 위와 같은 인공지능 작업의 일부 및 전부를 클라우드로 오프로딩하여 수행하는 것이다. 하지만, 클라우드 오프로딩 방법은 추가적인 지연시간을 초래하게 되는 단점이 있다.

본 논문에서는 인공지능 작업을 순수 드론 기반으로 수행하는 것과 순수 클라우드 기반으로 수행하는 것을 시뮬레이션/에뮬레이션 기반에서 비교 분석하는 것을 목표로 한다. 본 연구는 (i) 순수 드론 기반이 가지고 있는 처리용 한계와 (ii) 클라우드 기반이 가지고 있는 지연시간 한계 중 어느 요소가 드론 기반의 인공지능 작업 수행에서의 (목표물 추적 응용의 문맥에서) 성능에 얼마나 어떻게 영향을 주는지를 정량화하고, 좀 더 나은 목표물 추적 성능을 이루기 위해서 전체 시스템의 요소 중에서 어느 부분이 핵심 요소인지를 파악할 수 있는 근거를 제시할 것이라 기대한다. 이와 같은 결론은 물체인식 및 추적과 드론 제어에서 사용하는 인공지능 알고리즘의 특성에 의존적이며, 제한요소가 가미된 환경에서의 인공지능 알고리즘의 개선 및 선택의 기준을 파악하는데도 도움이 될 것이라고 생각한다. 마지막으로 앞으로 많이 고려되고 있는 클라우드 기반 인공지능 서비스에서 필요로 하는 통신망의 요구사항을 파악하는데도 큰 도움이 될 것이라

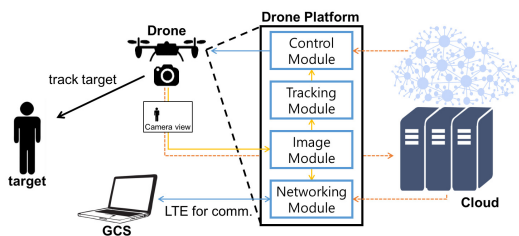


그림 1. 드론 기반 목표물 추적 시스템 개요
Fig. 1. Outline of the drone-based target tracking system

기대한다.

II. 관련 연구

목표물 추적 관련 연구: 추적시스템의 구현은 GPS 기반^{1,2}, 영상 센서 기반^{3,4}, 기타 센서 기반 (RF 신호기반^{5,6}, 음향신호 기반^{7,8})으로 분류할 수 있다. 위치기반의 경우, GPS가 필요하여 임의의 목표물 추적에는 적합하지 않다. 기타 센서 기반 시스템의 예로 RF 신호 또는 음향 신호를 사용한 시스템이 있으나, 잡음에 대한 성능 저하나 다중 반사와 같은 환경 변화에 민감한 단점이 있다. 영상 기반 시스템은 앞서 언급된 제약으로부터 자유로우면서 임의의 대상을 추적하기 쉬우나, 목표물 인식 및 추적을 위한 계산량이 많은 단점이 있다. 본 논문에서는 영상 기반의 추적 시스템을 고려하고, 계산량의 극복방안에 관하여 논의한다.

드론제어 방법 관련 연구: 드론제어영상 기반으로 목표물을 인식 후, 드론 제어를 위한 연구로는 PID (Proportional Integral Derivative) 제어 기반, MPC (Model Predictive Control) 기반, MPC + 강화학습기반 연구들이 있었다^{9,10}. 최근에는 깊은 신경망 기반 강화학습을 이용한 로봇 및 드론 제어에 관한 연구가 발표되고 있으며, 그 중 가장 성능이 좋다고 알려진 DDPG (Deep Deterministic Policy Gradient) 알고리즘¹¹을 활용한 드론 조종을 고려한다.

작업 클라우드 오프로딩 관련 연구: 드론 기반 목표물 추적 시스템과 같이 실시간성을 요구하는 응용에 대하여 많은 계산량을 요구하는 이미지 처리 부분을 클라우드로 오프로딩하여 성능 검증하는 연구가 존재한다¹². 이미지 처리를 위한 다양한 인공지능 알고리즘과, 입력 이미지 사이즈를 인자로 하여 처리시간과 성능 사이 트레이드 오프 관계를 살펴보았다. 또한 드론이 실시간으로 목표물을 찾는 응용을 위한 하이브리드 오프로딩 기법을 제안한 연구도 존재한다¹³. 이 방식은 드론에서 가벼운 인공지능 알고리즘을 통하여 인지 작업을 수행하고, 목표물이 인지 이벤트가 발생하며 클라우드로 고화질 영상을 보내서 보다 정확한 이미지 처리 결과를 통해 최종 목표물을 찾아낸다. 모바일 디바이스 환경에서 딥러닝 기반 증강현실 어플리케이션을 구동하기 위한 클라우드 오프로딩 방법론을 제시하였다¹⁴. 모바일 디바이스와 클라우드의 성능 비교를 바탕으로 어떤 상황에서 어떤 종류의

작업을 언제 클라우드로 오프로딩 해야 하는지에 대한 방법을 제시하고 있다. 본 논문에서는 순수 클라우드 기반과 순수 드론 기반의 환경에서의 추적시스템 전체를 모사하여, 다양한 인자에 대해서 추적성능의 변화를 정량화하는 것을 목표로 한다.

III. 인공지능 작업: 물체인식/추적 및 드론제어 알고리즘

3.1 물체인식 및 추적

연속된 영상으로부터 목표물의 위치와 크기 정보를 얻어내는 것이 목표이다. 그림 2에서 물체 인식 및 추적해 나가는 과정을 보여주고 있다.

알고리즘: 목표물 인식에서는 하나의 프레임 내에서 원하는 물체의 위치와 크기를 찾아내는 과정이다. 이를 위해 합성곱 신경망(CNN: Convolutional Neural Network) 기반의 알고리즘인 SSD¹⁵을 기용하였으며, SSD에서 공개한 미리 학습시켜 놓은 모델을 사용하였다. 이후의 여러 프레임에서는 물체 인식 정보를 바탕으로 물체가 전 프레임에서 현재 프레임으로 바뀌었을 때 움직임의 차를 계산하는데, 이를 위해서 물체 추적 알고리즘인 KCF¹⁶를 활용하였다. 그러나 물체 추적 알고리즘의 부정확성으로 인해 여러 프레임을 거쳐 갈수록 실제 물체의 위치와 위 알고리즘의 결과와 차이가 발생한다. 이를 보상하기 위해 처음 사용하였던 물체 인식 알고리즘을 주기적으로 사용하여 물체에 대한 정확한 정보를 갱신하였다.

성능: 물체 인식의 경우 한 프레임을 계산하는데 소요되는 시간이 증가할수록, 물체 추적에 제공하는 정보가 실제 물체 위치와 정확하지 않아 전체적인 성능이 감소한다. 물체 추적의 경우도 이와 유사하게, 주어진 시간 내에 많은 프레임을 처리할수록 물체 이동 경로를 보다 정확하게 계산할 수 있다.

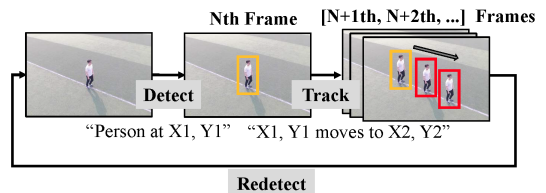


그림 2. 물체 인식 및 추적 절차
Fig. 2. The target detection and tracking process

3.2 드론 제어 알고리즘

드론 기반 목표물 제어 시스템에서 드론 제어 알고리즘에서는 목표물 인식 및 추적 과정에서 얻은 물체의 위치, 크기 등의 정보를 이용하여 드론의 움직임의 결정을 책임진다. 물체 움직임의 불확실성, 드론의 물리적 특성, 장애물의 존재 가능성 등의 원인에 의해 최적의 움직임을 만들어 내기 위해서 강화학습 기반 드론 제어 알고리즘을 사용한다.

강화학습은 주어진 환경에서 어떤 행동을 해야 결과적으로 가장 큰 보상을 얻을 수 있을지를 연속적인 행동의 선택을 통해 학습하는 과정이다. 본 연구에서는 그림 3과 같이 로봇 및 이동체의 움직임 제어에 적합하다고 알려져 있는 DDPG (Deep Deterministic Policy Gradient) 기법을 이용하였다.

DDPG는 Actor/Critic 알고리즘, DPG (Deterministic policy gradient), 그리고 딥러닝 모듈을 구성요소로 하고 있다. Actor-Critic 알고리즘에서는, 상태를 보고 무슨 행동을 할지 결정하는 Actor와 Actor의 행동을 평가하는 Critic을 분리하여 학습한다. DPG는 연속적인 행동 공간에서 정밀한 행동을 학습하기 위해서 어떻게 학습해야하는지에 대한 수학적 과정을 제공한다. 마지막으로 딥러닝을 사용하여 복잡성이 높은 모델을 정확성을 확보하면서, 적절한 계산량으로 학습을 가능하게 한다. DDPG 기법은 이러한 여러 모듈들의 상호작용을 통해 복잡한 상황에 대해 보상을 최대화하는 방향으로 정밀한 제어를 가능하게 한다. 행동을 평가하는 critic network는 weights θ^Q 와 주어진 state, action에 대해 $Q(s, a|\theta^Q)$ 로 주어지며, 행동을 결정하는 actor network는 weights θ^μ 에 대해 $\mu(s|\theta^\mu)$ 로 주어진다. 전체 학습은 이러한 네트워크 간의 상호작용의 반복을 통해 얻어진 샘플들을 통해 실시간으로 이루어진다. 먼저 critic network를 학습하

기 위하여 샘플에서 얻어진 보상 함수 값들을 사용하여 업데이트한다. 그 후 critic network를 통해 actor network를 평가하여 actor network를 업데이트한다. 이러한 critic network의 사용은 actor network를 단순히 보상 함수만을 사용하여 업데이트 하는 것보다 안정적으로 만들어준다. 또한 critic network, actor network보다 느리게 학습하는 target network $Q(s, a|\theta^Q)$ 과 $\mu(s|\theta^Q)$ 을 추가적으로 사용하여 학습을 안정화 시킨다.

DDPG에서 정의한 상태(state)와 액션(action)의 종류와 보상함수는 다음과 같이 설정하였다.

- **상태:** 목표물의 드론 화면상 위치 및 크기, 드론의 속도 및 높이 히스토리,
- **행동:** 드론의 움직임을 제어를 위한 앞뒤, 좌우, 상하, 및 보는 방향
- **보상 함수:** 추적 정확도와 에너지 효율간의 Weighted 합. 추적 정확도는 화면상의 목표물 위치 및 크기로 정량화 됨.

물체의 움직임을 정확하게 추적 및 예측하기 위하여 한 지점의 위치가 아닌 움직임의 시간적 연관성의 학습을 고려하였다. 행동은 드론의 모든 가능한 물리적 움직임을 포함하였고, 물리적으로 가능한 모든 움직임이 가능하게 하였고, 보상 함수는 추적의 정확도와 드론의 에너지의 효율을 트레이드오프 할 수 있도록 설계하였다. 보상함수의 디자인에 따라 인공지능 학습 효율, 응용의 구체적인 특성 (드론의 에너지 효율, 추적 성능 등)이 달라지게 된다. 응용의 구체적인 특성이 주어지지 않았기 때문에 인공지능 학습 효율에 중점을 두어 보상 함수를 결정하였다.

IV. 시뮬레이션/에뮬레이션 환경

4.1 순수 드론 기반 vs. 순수 클라우드 기반

본 논문에서는 인공지능 작업을 순수 드론 기반으로 수행하는 것과 순수 클라우드 기반으로 수행하는 것을 비교 분석하는 것을 목표로 한다. 이를 위하여 그림 4와 같은 시뮬레이션/에뮬레이션 환경을 구성하였다. 그림 4(a)에서 보여주듯이 순수 드론 기반의 경우, 카메라로부터 얻은 영상을 처리하여 드론의 움직임을 결정하는 과정을 모두 드론에서 수행한다. 이 경우 드론에서의 컴퓨팅 능력의 한계로, 물체인식 및 추

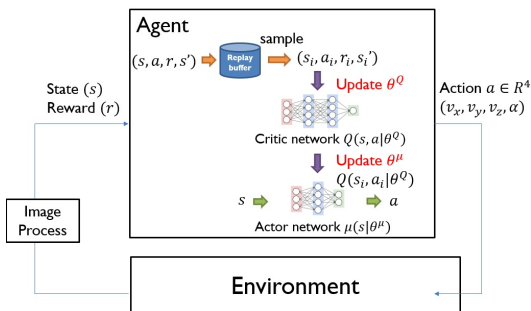


그림 3. DDPG 알고리즘의 절차
Fig. 3. Working mechanism of DDPG

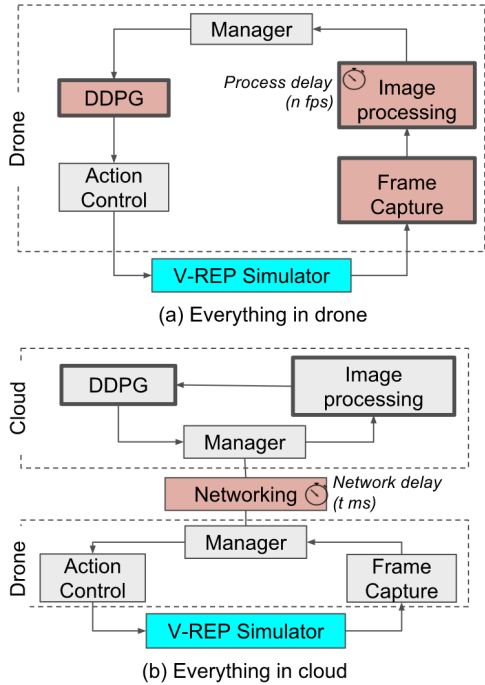


그림 4. 시뮬레이션 환경: 드론 기반 vs. 클라우드 기반
 Fig. 4. Comparison between the drone-based and cloud-based simulation environments

적을 위한 처리율에 한계가 시스템 성능을 제한할 수 있다. 그림 4(b)와 같이 순수 클라우드 기반 시스템의 경우, 이미지 프레임 캡처만 드론에서 이루어지고, 이미지는 네트워크를 거쳐서 클라우드로 전송되어 인식, 처리, 드론 제어의 모든 과정이 클라우드에서 이루어진다. 따라서 이미지 및 명령 전달을 위한 네트워크에서의 전송시간이 시스템 성능을 제한할 수 있다. 우리의 관심사는 어떠한 조건에서 어느 방식의 시스템이 어느 정도의 성능을 보여주는가를 정량적으로 분석 및 비교하는 것이다.

V-REP 기반 시뮬레이터: 본 논문에서 보고자 하는 환경변화에 따른 실제 드론의 움직임을 모사하기 위해서 V-REP (Virtual Robot Experimentation Platform)을 수정하여 제작하였다. 를 기반으로 시뮬레이션을 구성하였다. V-REP는 가상 환경 로봇 시뮬레이터로 각종 API를 통해 Object/Model 을 개별적으로 제어 가능하고, API를 다양한 프로그래밍 언어 (C/C++, Python, Java 등)로 제공한다. 또한 다양한 센서와 액추에이터 모델을 제공하며, 로봇, 드론, 자동차들에 센서와 액추에이터 모델들을 추가하여 다양한 시뮬레이션을 수행할 수 있다. 본 연구에서 확장한 부

분은 카메라 센서를 부착한 드론과, 목표물 역할을 하는 물체로 구성된다. Python 언어로 제작된 시뮬레이션 코드는 API를 통하여 드론을 조종하며, 카메라로부터 실시간 영상을 받아서 처리하는 과정을 거치게 된다. 또한 목표물 물체를 임의로 움직이게 하여 드론이 목표물을 추적하는 시뮬레이션을 진행하였다.

주요 환경 인자: 순수 드론기반 시스템과 순수 클라우드기반 시스템의 성능 차이를 분석하기 위해서, 순수드론기반에서는 이미지 프레임 처리율, 순수클라우드에서는 네트워크 지연시간을 주요환경인자로 사용하고, 전체적인 실험에서는 목표물의 이동 속도를 변화시켜 가면서, 성능을 조사한다.

시뮬레이션/에뮬레이션 환경 구성: 드론기반 인공지능 작업환경을 모사하기 위해서, 물체인식 및 추적을 위한 알고리즘의 수행시간을 실제 실험을 바탕으로 기본 인자 설정을 진행하였다. 드론에 올려서 사용할 수 있는 컴퓨팅 보드인 NVIDIA Jetson TX2 보드에서는 위의 인공지능 작업을 수행하는데 약 7 fps의 속도를 가진다 (카메라의 이미지 캡처 속도는 30 fps). 순수 클라우드 기반 시스템에서는 실제 이더넷 네트워크를 활용하여, 고성능 클라우드용 컴퓨터(Intel Xeon CPU: 8 cores, 3.70Ghz, TITAN Xp with 3840 GPU, 16GB RAM, 12GB GPU memory)와 연결하여, 처리하도록 시스템을 구성하였고, 실험적으로 30 fps를 충분히 처리할 수 있음을 확인하였다. 유선 이더넷 네트워크 인터페이스에 TC (Traffic Control) 모듈을 적용하여, 다양한 지연시간을 일부러 발생시켜서 실험할 수 있도록 하였다. 또한 실제 추적 환경을 기반으로 하여 보상 함수, 추적 정확도 등에서 사용되는 상수를 결정하였다.

V. 결과 및 해석

5.1 성능 평가 기준

제안하는 시스템의 성능 평가 기준은 드론의 목표물 추적 정확도이다. 영상 추적 분야에서는 영상 추적 정확도는 다양한 형태의 표준화 된 평가 기준이 존재한다^[17]. 본 논문에서는 다음과 같은 두 기준에 따라 드론 추적 시스템의 성능을 측정 및 평가하였다.

5.1.1 Tracking accuracy

64x64 이미지 상에서 추적 물체와 화면 중앙과의 x 축 거리를 dx , y축 거리를 dy , 현재 시간을 t 라고 했

을 때, 현재 추적 정확도 $A(t)$ 는 다음과 같은 형태로 계산 된다.

$$A(t) = \begin{cases} 1 & \text{if } dx < 8, dy < 8 \\ 0.5 & \text{else if } dx < 16, dy < 16 \\ 0 & \text{Otherwise} \end{cases}$$

추적 정확도 측정에서의 기본 인자들은 실제 드론 활용 목표물 추적 응용을 바탕으로 설정 하였다. 현재 추적 정확도 $A(t)$ 를 이용하여 평균 추적 정확도 A_{avr} 를 측정한다.

5.1.2 Fail count

위의 경우 물체를 놓치지 않고 추적했을 때를 가정 하고 성능을 측정하기 때문에 물체가 추적 물체를 자주 놓칠 경우 추적 시스템의 성능 측정이 비교적 정확 하지 않을 수 있다. 따라서 목표 물체를 놓치는 횟수를 측정하여 성능을 평가한다. 현재 시간을 t 라고 했을 때, 시간 t 에서 화면 상 목표 물체 존재 여부를 $M(t)$ 로 정의한다. 화면에 목표 물체가 존재 한다면 $M(t) = 0$ 이며, 화면에 목표 물체가 존재하지 않는다면 $M(t) = 1$ 이다. $M(t)$ 에 대해 Fail count $F(t)$ 를 다음과 같이 정의한다.

$$F(t) = F(t - 1) + M(t)$$

최종 fail count $F(T)$ 를 이용하여 성능을 측정한다.

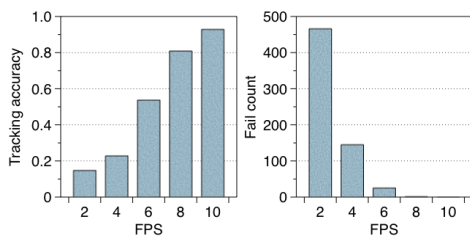


그림 5. 이미지 프레임 처리율에 따른 추적 성능 평가
Fig. 5. Tracking accuracy for varying fps

5.2 실험 결과

본 논문에서는 드론 추적 시스템에서 주어지는 3가지 환경 요소인 (i) 컴퓨팅 파워, (ii) 네트워크 지연시간, (iii) 목표물 속도에 따라 추적 정확도가 어떻게 달라지는 지 분석한다.

이미지 프레임 처리율의 효과. 드론 기반 목표물

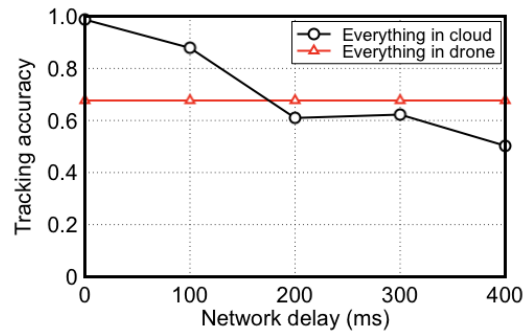


그림 6. 네트워크 지연시간에 따른 추적 정확도
Fig. 6. Tracking accuracy for varying network delay

추적 시스템에서 컴퓨팅 파워는 인공지능 작업 수행 능력에 큰 영향을 끼친다. 실제 실험을 바탕으로 컴퓨팅 보드의 GPU, CPU, 메모리 사용률을 측정한 결과 이미지 처리 부분에서 대부분을 차지함을 확인하였다. 이를 통해 장치가 가진 컴퓨팅 파워를 이미지 프레임 처리율로 생각하여 이에 따른 목표물 추적 성능을 평가하였다. 그림5에서는 목표물은 1m/s로 움직이는 상황에서 이미지 프레임 처리율이 목표물 추적 성능에 미치는 영향에 대해 살펴본다. 이미지 프레임 처리율이 증가할수록 드론의 목표물 추적 성능이 향상됨을 알 수 있다. 이는 드론을 더 짧은 간격으로 섬세하게 조종하여 목표물 추적할 수 있기 때문이다. 순수 드론 기반 시스템의 경우 제한된 컴퓨팅 파워 때문에 이미지 프레임 처리율에 제약이 있다. 하지만 클라우드를 활용하면 이미지 프레임 처리율 증가로 인한 추적 성능 향상을 기대할 수 있다.

네트워크 지연시간의 효과. 순수 클라우드 기반 시스템의 경우 높은 컴퓨팅 파워로 최대의 이미 프레임 처리율인 30 fps를 달성할 수 있다. 하지만 클라우드와 드론간 네트워크 지연시간으로 인한 추적 성능 저하가 예상된다. 따라서 네트워크 지연시간이 추적 성능에 미치는 영향을 살펴보고자 한다. 그림6에서 알 수 있듯이 네트워크 지연시간이 커질수록 추적 성능이 감소하며, 지연시간이 200ms 이상되면 7 fps의 이미지 프레임 처리율을 갖는 순수 드론 기반 시스템보다 낮은 성능을 보인다. 따라서 네트워크의 상태에 따라 클라우드의 사용여부를 잘 결정하여 사용하여야 한다.

목표물 속도에 따른 추적 성능 변화. 목표물 추적의 성능은 목표물의 움직임 특성에도 영향을 받는다.

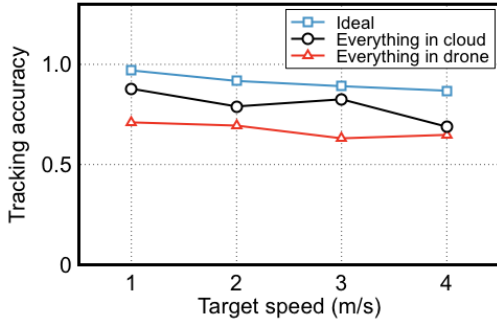


그림 7. 목표물의 이동 속도에 따른 추적 정확도
Fig. 7. Tracking accuracy for varying target speed

그림 7에서는 목표물이 움직이는 속도에 따라서 순수 드론 기반 시스템과 순수 클라우드 기반 시스템의 추적 성능을 보여준다. 클라우드의 네트워크 지연시간은 100ms로 설정하였으며, 드론의 이미지 프레임 처리율은 5 fps로 설정하였다. 달성 가능한 최대 성능인 네트워크 딜레이가 없으며 30 fps 이미지 처리율을 가지는 경우 (Ideal)와의 비교를 통하여 각 시스템의 특징을 알아보았다. 순수 클라우드 기반 시스템의 경우 낮은 속도에서는 이상적인 경우와 비슷한 성능을 달성하지만 속도가 증가함에 따라 성능이 급격하게 나빠지는 경향을 보인다. 이에 반해 순수 드론 기반 시스템의 경우 이상적인 경우와 큰 성능차이를 보이지만 목표물의 속도가 증가함에 따라 적은 변화를 보여준다.

VI. 결론 및 추후과제

목표물 추적의 성능은 드론 또는 클라우드가 가지는 컴퓨팅 파워, 드론과 클라우드간 네트워크 지연시간, 그리고 목표물의 이동 특성에 따라 결정된다. 드론 목표물 추적과 같이 실시간성을 필요로 하는 응용의 경우 인공지능 작업의 오프로딩이 상황에 따라 적절이 이루어 질 때 최적의 성능을 달성할 수 있다. 또한 클라우드를 사용함에 있어 가장 중요한 요소는 네트워크 지연시간이다. 5G의 목표 중 하나인 저지연 네트워크가 달성되면 실시간성이 중요한 응용에서도 클라우드가 더 활발히 쓰일 수 있을 것이며 동시에 응용의 성능 향상을 기대할 수 있다.

이 연구에서 우리는 제한된 컴퓨팅 성능을 가진 드론, 드론과 클라우드 간의 제한된 네트워크 성능이 존재하는 환경에서 무인 항공기 기반 목표물 추적 시스템의 성능을 측정하였다. 목표물 추적의 성능은 드론 또는 클라우드가 가지는 컴퓨팅 파워, 드론과 클라우

드론 네트워크 지연시간, 그리고 목표물의 이동 특성에 따라 결정된다. 드론 목표물 추적과 같이 실시간성을 필요로 하는 응용의 경우 인공지능 작업의 오프로딩이 상황에 따라 적절이 이루어 질 때 최적의 성능을 달성할 수 있다. 또한 클라우드를 사용함에 있어 가장 중요한 요소는 네트워크 지연시간이다. 5G의 목표 중 하나인 저지연 네트워크가 달성되면 실시간성이 중요한 응용에서도 클라우드가 더 활발히 쓰일 수 있을 것이며 동시에 응용의 성능 향상을 기대할 수 있다. 추후 과제로는 복잡한 계산을 요구하는 인공지능 작업 중 일부 작업은 드론에서 일부 작업은 클라우드에서 수행하여 최적의 성능을 달성할 수 있는 기법에 대한 연구가 필요하다. 작업을 나누는 일 또한 인공지능을 적용하여 최적의 성능을 달성할 수 있는 방법에 대한 연구가 필요하다.

References

- [1] Top 10 drones with follow me mode in 2016, <http://www.top10drone.com/top-10-drones-follow-me-mode/>.
- [2] 12 best follow me drones and follow me technology reviewed, <https://www.dronezon.com/drone-reviews/best-follow-me-gps-mode-drone-technology-reviewed/>.
- [3] J.-E. Gomez-Balderas, G. Flores, L. G. Carrillo, and R. Lozano, "Tracking a ground moving target with a quadrotor using switching control," *J. Intell. & Robotic Syst.*, vol. 70, no. 1-4, pp. 65-78, 2013.
- [4] J. Kim and D. H. Shim, "A vision-based target tracking control system of a quadrotor by using a tablet computer," *ICUAS*, pp. 1165-1172, 2013.
- [5] T. Wei and X. Zhang, "mTrack: high precision passive tracking using millimeter wave radios," in *Proc. ACM MobiCom*, pp. 117-129, Paris, France, Sept. 2015.
- [6] F. Adib, Z. Kabelac, and D. Katabi, "Multi-person localization via RF body reflections," in *12th USENIX Symp. NSDI 15*, pp. 279-292, Oakland, CA, May 2015.
- [7] W. Mao, J. He, and L. Qiu, "CAT: high-precision acoustic motion tracking," in *Proc. ACM MobiCom*, 2016.

[8] R. Nandakumar, V. Iyer, D. Tan, and S. Gollakota, "FingerIO: Using active sonar for fine-grained finger tracking," in *Proc. ACMCHI*, pp. 1515-1525, 2016.

[9] R. Barták and A. Vykovský, "Any object tracking and following by a flying drone," *IEEE MICA*, pp. 35-41, Oct. 2015.

[10] A. Hernandez, H. Murcia, C. Copot, and R. De Keyser, "Model predictive path-following control of an AR. Drone quadrotor," in *Proc. CLCAÁŠ14*, pp. 14-17, Cancun, Quintana Roo, Mexico, 2014.

[11] T. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra. "Continuous control with deep reinforcement learning," *Computer Sci.*, 2017.

[12] H. Genc, T. Chin, M. Halpern, and V. J. Reddi, "Optimizing sensor-cloud architectures for real-time autonomous drone operation," in *Proc. Sensors to Cloud Architectures Wksp.*, 2017.

[13] J. Lee, J. Wang, D. Crandall, S. Sabanovic, and G. Fox, "Real-time, cloud-based object detection for unmanned aerial vehicles," *IEEE Int. Conf. Robotic Computing*, pp. 36-43, Apr. 2017.

[14] X. Ran, H. Chen, Z. Liu, and J. Chen, "Delivering deep learning to mobile devices via offloading," in *Proc. Wksp. Virtual Reality and Augmented Reality Network*, pp. 42-47, 2017.

[15] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed, "SSD: Single shot multibox detector," in *ECCV*, 2016.

[16] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," in *Proc. IEEE Trans. Patter Anal. Mach. Intell.*, pp. 583-596, 2015.

[17] S. Oh, "A case study civilian drone," in *Proc. KSBE Summer Conf.*, 2015.

[18] L. Čehovin, A. Leonardis, and M. Kristan, "Visual object tracking performance measures revisited," *IEEE Trans. Image Process*, vol. 25, no. 3, pp. 1261-1274, 2016.

손 경 환 (Kyunghwan Son)



2017년 2월 : 한국과학기술원
전기 및 전자 공학과 졸업
2017년 2월~현재 : 한국과학기술원
전기 및 전자 공학과
석사과정
<관심분야> 강화학습, 딥러닝,
드론, 클라우드

David Hostallero



2016년 6월 : University of the
Philippines Diliman (BS
Computer Science) 졸업
<관심분야> 강화학습, 컴퓨터
비전, 데이터 마이닝

김 대 우 (Daewoo Kim)



2013년 2월 : 연세대학교 전기
전자공학과 졸업
2013년 3월~현재 : 한국과학기술원
전기 및 전자 공학과
석박사 통합 과정
<관심분야> 강화학습, 딥러닝,
IoT

방 지 환 (Jihwan Bang)



2017년 2월 : 한국과학기술원
전기 및 전자공학과 졸업
2017년 3월~현재 : 한국과학기술원
전기및전자공학과 석사
<관심분야> 강화학습, 딥러닝,
UAV 시스템, 네트워크, 클
라우드

강 완 주 (Wan Ju Kang)



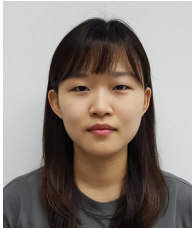
2017년 8월 : 한국과학기술원
전기 및 전자공학과 졸업
2017년 9월~현재 : 한국과학기술원
전기및전자공학과 석사
<관심분야> 강화학습, 딥러닝,
네트워킹, IoT

하 재 형 (Jae-hyeong Ha)



2017년 2월 : 한국과학기술원
전기 및 전자 공학부 학사
졸업
2017년 3월~현재 : 한국과학기술원
전기 및 전자공학부
석사 과정
<관심분야> 컴퓨터 네트워크,
블록체인

윤 세 은 (Se-eun Yoon)



2014년 3월~현재 : 한국과학기술원
전기 및 전자공학과 학사
<관심분야> 강화학습, 딥러닝,
네트워킹, IoT

한 동 수 (Dongsu Han)



2004년 2월 : KAIST 전산학부
(학사)
2010년 12월 : Carnegie Mellon
University 전산학부 (석사 -
석박연계)
2012년 12월 : Carnegie Mellon
University 전산학부 (박사 -
석박연계)

구 윤 표 (Yoon-pyo Koo)



2016년 2월 : 한국과학기술원
전기및전자공학과 졸업
2016년 3월~현재 : 한국과학기술원
전기및전자공학과 석사
<관심분야> 강화학습, 딥러닝,
다중 UAV 시스템

2013년 1월~2013년 5월 : Carnegie Mellon University
전산학부 (박사 후 연구원)
2013년 6월~현재 : KAIST 전기전자공학부 교수
2013년 10월~현재 : KAIST 정보보호대학원 겸임교수
<관심분야> 네트워크 시스템 디자인, 인터넷/시스템
보안

여 현 호 (Hyun-ho Yeo)



2017년 2월 : 한국과학기술원
전기 및 전자 공학부 학사
졸업
2017년 3월~현재 : 한국과학기술원
전기 및 전자공학부
석사 과정
<관심분야> 컴퓨터 네트워크,
딥러닝

이 융 (Yung Yi)



1997년 2월 : 서울대 컴퓨터공
학부 (학사)
1999년 2월 : 서울대 컴퓨터공
학부 (석사)
2006년 8월 : University of
Texas at Austin, USA 전기
컴퓨터공학부 (박사)
2008년 8월~현재 : KAIST 전기전자공학부 교수
2009년 3월~2014년 8월 : KAIST 전산학부 겸임교수
2006년 8월~2008년 8월 : Princeton Univ. 전기전자
공학부(박사 후 연구원)
<관심분야> 무선통신네트워크, 머신러닝