



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2025-0012822
(43) 공개일자 2025년01월31일

(51) 국제특허분류(Int. Cl.)
G01S 11/06 (2006.01) G01S 1/04 (2006.01)
(52) CPC특허분류
G01S 11/06 (2013.01)
G01S 1/045 (2013.01)
(21) 출원번호 10-2023-0092834
(22) 출원일자 2023년07월18일
심사청구일자 2023년07월18일

(71) 출원인
한림대학교 산학협력단
강원도 춘천시 한림대학길 1, 한림대학교(옥천동)
(72) 발명자
김의직
경기도 남양주시 다산중앙로146번길 7, 2204동
1902호(다산동, 다산롯데캐슬)
권정혁
강원도 춘천시 남춘로5번길 14-1, 301호(퇴계동)
(74) 대리인
특허법인본

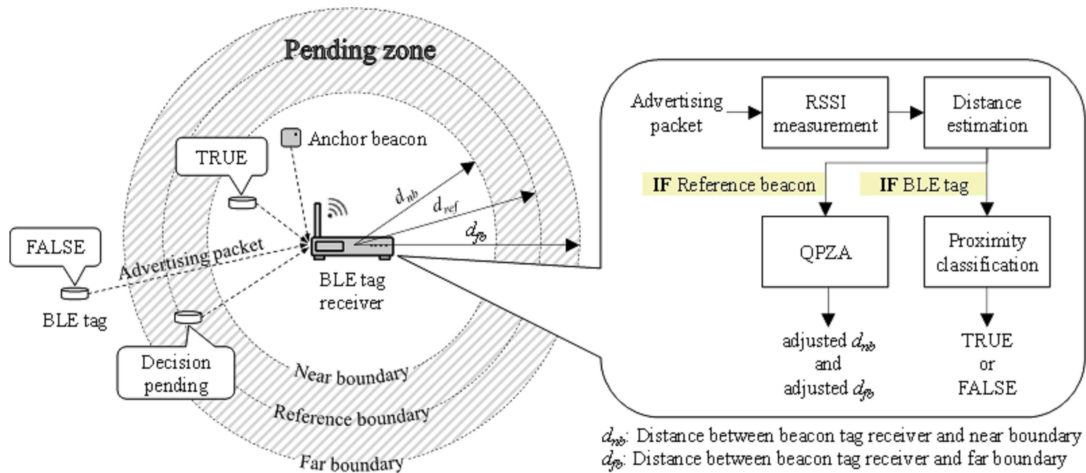
전체 청구항 수 : 총 10 항

(54) 발명의 명칭 노이즈 레벨에 기초한 경계 설정을 통해 근접 감지를 수행하는 Tag Receiver

(57) 요약

Tag Receiver의 제어 방법이 개시된다. 본 제어 방법은, Tag로부터 식별 정보를 포함하는 제1 신호를 수신하는 단계, 수신된 제1 신호를 기반으로 Tag에 대한 제1 거리를 추정하는 단계, 일정한 지점에 위치하는 Anchor Beacon으로부터 주기적으로 제2 신호를 수신하는 단계, 수신된 제2 신호를 바탕으로 Anchor Beacon에 대한 제2 거리를 추정하는 단계, Tag Receiver와 Anchor Beacon 간의 실제 거리 및 추정된 제2 거리를 기초로 노이즈 레벨을 획득하는 단계, 노이즈 레벨을 기초로 근거리 경계 및 원거리 경계를 설정하는 단계, 근거리 경계 및 상기 원거리 경계 중 적어도 하나를 추정된 제1 거리와 비교하여, Tag의 사용자의 Tag Receiver에 대한 근접 여부를 판단하는 단계를 포함한다.

대표도



이 발명을 지원한 국가연구개발사업

과제고유번호	1425171710
과제번호	S3278476
부처명	중소벤처기업부
과제관리(전문)기관명	중소기업기술정보진흥원
연구사업명	중소기업기술혁신개발사업(시장확대형)
연구과제명	무선 태그 및 레이더를 활용한 IoT 융합형 보행자 및운전자 안전 솔루션 개발
기 여 율	1/2
과제수행기관명	엠투테크 주식회사
연구기간	2022.05.01 ~ 2024.04.30

이 발명을 지원한 국가연구개발사업

과제고유번호	1711181469
과제번호	2021R1C1C2095696
부처명	과학기술정보통신부
과제관리(전문)기관명	한국연구재단
연구사업명	개인기초연구(과기정통부)
연구과제명	의료 사물인터넷을 위한 부하분산 블록엣지 프레임워크개발
기 여 율	1/2
과제수행기관명	한림대학교
연구기간	2021.09.01 ~ 2026.02.28

명세서

청구범위

청구항 1

Tag Receiver의 제어 방법에 있어서,

상기 Tag Receiver가, Tag로부터 식별 정보를 포함하는 제1 신호를 수신하는 단계;

상기 Tag Receiver가, 상기 수신된 제1 신호를 기반으로 상기 Tag에 대한 제1 거리를 추정하는 단계;

상기 Tag Receiver가, 일정한 지점에 위치하는 Anchor Beacon으로부터 주기적으로 제2 신호를 수신하는 단계;

상기 Tag Receiver가, 상기 수신된 상기 제2 신호를 바탕으로 상기 Anchor Beacon에 대한 제2 거리를 추정하는 단계;

상기 Tag Receiver가, 상기 Tag Receiver와 상기 Anchor Beacon 간의 실제 거리 및 상기 추정된 제2 거리를 기초로 노이즈 레벨을 획득하는 단계;

상기 Tag Receiver가, 상기 노이즈 레벨을 기초로 근거리 경계 및 원거리 경계를 설정하는 단계; 및

상기 Tag Receiver가, 상기 근거리 경계 및 상기 원거리 경계 중 적어도 하나를 상기 추정된 제1 거리와 비교하여, 상기 Tag의 사용자의 상기 Tag Receiver에 대한 근접 여부를 판단하는 단계;를 포함하는, Tag Receiver의 제어 방법.

청구항 2

제1항에 있어서,

상기 제2 거리를 추정하는 단계는,

상기 Anchor Beacon으로부터 상기 Tag Receiver로 전송된 상기 제2 신호의 RSSI 값에 대응되는, 상기 Tag Receiver와 상기 Anchor Beacon 간의 추정 거리를 식별하는, Tag Receiver의 제어 방법.

청구항 3

제2항에 있어서,

상기 Tag Receiver는,

상기 근거리 경계의 상태 별로 다음 액션을 선택하기 위한 Q-value가 저장된 제1 Q-table; 및

상기 원거리 경계의 상태 별로 다음 액션을 선택하기 위한 Q-value가 저장된 제2 Q-table;을 포함하고,

상기 근거리 경계 및 원거리 경계를 설정하는 단계는,

Q-러닝을 기반으로 하는 제1 에이전트에 상기 노이즈 레벨을 입력하여 Q-value를 획득하고, 상기 획득된 Q-value를 상기 제1 Q-table에 적용하여 상기 근거리 경계의 다음 상태를 판단하고,

Q-러닝을 기반으로 하는 제2 에이전트에 상기 노이즈 레벨을 입력하여 Q-value를 획득하고, 상기 획득된 Q-value를 상기 제2 Q-table에 적용하여 원거리 경계의 다음 상태를 판단하는, 근접 감지 시스템.

청구항 4

제3항에 있어서,

상기 Tag Receiver의 제어 방법은,

이하 수학적식에 따라 노이즈 레벨에 대한 데이터 세트를 정의하는 단계; 및

$$\mathbf{NL} = \{\sigma_1, \sigma_2, \dots, \sigma_i\}$$

이하 수학식에 따라 상기 NL에 대한 밀도 함수를 획득하는 단계;를 포함하고,

$$f(x) = \frac{1}{\sum_{i=1}^I h_i} \sum_{i=1}^I K\left(\frac{x - \sigma_i}{h_i}\right)$$

h 는 KDE의 대역폭이며, $K(x)$ 는 $K(x) = e^{-x^2/2}$ 로 주어지는 가우시안 커널 함수이고,

상기 Tag Receiver의 제어 방법은,

상기 밀도 함수의 로컬 최대값의 수에 따라 클러스터의 수를 결정하는 단계; 및

상기 클러스터의 수를 바탕으로 노이즈 레벨을 통해 선택되는 상태의 수를 결정하는 단계;를 포함하는, Tag Receiver의 제어 방법.

청구항 5

제3항에 있어서,

상기 근거리 경계 및 원거리 경계를 설정하는 단계는,

이하 수학식에 따라 상기 제1 에이전트에 의해 선택된 액션을 식별하고,

$$a_t^{nb} = \arg \max_a Q(s_t^{nb}, a)$$

이하 수학식에 따라 상기 근거리 경계에 대한 다음 상태를 결정하고,

$$s_{t+1}^{nb} = \begin{cases} s_t^{nb} - 1 & a_t^{nb} = 1, s_t^{nb} > 1 \\ s_t^{nb} + 1 & a_t^{nb} = 3, s_t^{nb} < m \\ s_t^{nb} & \text{otherwise} \end{cases}$$

이하 수학식에 따라 상기 제2 에이전트에 의해 선택된 액션을 식별하고,

$$a_t^{fb} = \arg \max_a \hat{Q}(s_t^{fb}, a)$$

이하 수학식에 따라 상기 원거리 경계에 대한 다음 상태를 결정하는,

$$s_{t+1}^{fb} = \begin{cases} s_t^{fb} - 1 & a_t^{fb} = 1, s_t^{fb} > 1 \\ s_t^{fb} + 1 & a_t^{fb} = 3, s_t^{fb} < m \\ s_t^{fb} & \text{otherwise} \end{cases}$$

Tag Receiver의 제어 방법.

청구항 6

제5항에 있어서,

상기 근거리 경계 및 원거리 경계를 설정하는 단계는,

상기 근거리 경계 및 상기 원거리 경계 각각에 대하여 결정된 다음 상태를 바탕으로, 상기 근거리 경계 및 상기 원거리 경계 각각이 조정된 값인 d'_{nb} 및 d'_{fb} 를 획득하고,

상기 Tag Receiver의 제어 방법은,

이하 수학식에 따라 노이즈 레벨(σ) 및 상기 조정된 값을 비교하여 상기 제1 에이전트 및 상기 제2 에이전트 각각과 관련된 보상을 산출하는 단계;를 포함하고,

$$r_t^{nb} = \begin{cases} -1 & |d_{ref} - d'_{nb}| < \sigma/k \\ 0 & |d_{ref} - d'_{nb}| = \sigma/k \\ +1 & |d_{ref} - d'_{nb}| > \sigma/k \end{cases}$$

$$r_i^{\text{ref}} = \begin{cases} +1 & |d_{i,\text{ref}}^{\text{r}} - d_{i,\text{ref}}^{\text{a}}| < \sigma/k \\ 0 & |d_{i,\text{ref}}^{\text{r}} - d_{i,\text{ref}}^{\text{a}}| = \sigma/k \\ -1 & |d_{i,\text{ref}}^{\text{r}} - d_{i,\text{ref}}^{\text{a}}| > \sigma/k \end{cases}$$

K는 노이즈 레벨의 계수이고, d_{ref} 는 기준 거리인, Tag Receiver의 제어 방법.

청구항 7

제1항에 있어서,

상기 제1 거리를 추정하는 단계 및 상기 제2 거리를 추정하는 단계 중 적어도 하나는,

RSSI 값을 바탕으로 거리 추정을 수행하기 위한 DNN(Depp Neural Networks) 모델로, 상기 Tag로부터 순차적으로 수신되는 복수의 신호 각각의 RSSI 값을 입력하고,

상기 DNN 모델은,

RSSI 값을 다중으로 입력 받기 위한 복수의 노드를 포함하는 입력 레이어;

상기 입력 레이어의 출력과 연결되는 복수의 히든 레이어; 및

상기 복수의 히든 레이어와 연결되고, 추정 거리를 출력하기 위한 하나의 노드를 포함하는 출력 레이어;를 포함하는, 근접 관리 시스템.

청구항 8

적어도 하나의 인스트럭션이 저장된 메모리;

Tag 및 Anchor Beacon과 각각 통신을 수행하기 위한 통신부; 및

상기 인스트럭션을 수행하여 제1항의 제어 방법을 수행하는 프로세서;를 포함하는, Tag Receiver.

청구항 9

Tag Receiver의 프로세서에 의해 실행되어 상기 Tag Receiver로 하여금 제1항의 제어 방법을 수행하도록 하는, 컴퓨터 프로그램이 저장된 비일시적 컴퓨터 판독 가능 매체.

청구항 10

RSSI 값을 바탕으로 거리 추정을 수행하기 위한 DNN(Depp Neural Networks) 모델을 포함하는, Tag Receiver의 제어 방법에 있어서,

상기 Tag Receiver가, 상기 Tag로부터 순차적으로 수신되는 복수의 제1 신호 각각의 RSSI 값을 상기 DNN 모델로 입력하는 단계;

상기 Tag Receiver가, 상기 DNN 모델의 출력을 바탕으로 상기 Tag에 대한 제1 거리를 추정하는 단계;

상기 Tag Receiver가, 일정한 지점에 위치하는 Anchor Beacon으로부터 주기적으로 제2 신호를 수신하는 단계;

상기 Tag Receiver가, 상기 수신된 상기 제2 신호를 바탕으로 상기 Anchor Beacon에 대한 제2 거리를 추정하는 단계;

상기 Tag Receiver가, 상기 Tag Receiver와 상기 Anchor Beacon 간의 실제 거리 및 상기 추정된 제2 거리를 기초로 노이즈 레벨을 획득하는 단계;

상기 Tag Receiver가, 상기 노이즈 레벨을 기초로 근거리 경계 및 원거리 경계를 설정하는 단계; 및

상기 Tag Receiver가, 상기 근거리 경계 및 상기 원거리 경계 중 적어도 하나를 상기 추정된 제1 거리와 비교하여, 상기 Tag의 사용자의 상기 Tag Receiver에 대한 근접 여부를 판단하는 단계;를 포함하는, Tag Receiver의 제어 방법.

발명의 설명

기술분야

[0001] 본 개시는 Tag Receiver에 관한 것으로, 보다 상세하게는, RSSI(Received Signal Strength Indicator) 기반의 근접 분류(Proximity Classification)를 위하여 Q-learning 기반의 보류 구간(pending zone) 조정을 수행하는 Tag Receiver에 관한 것이다.

배경기술

[0002] 일반적으로 RSSI(Received Signal Strength Indicator)는 근접성 분류를 위해 장치 간의 거리를 추정하는 데 사용된다. 구체적으로, 수신자는 발신자가 전송한 신호의 RSSI를 측정하고 RSSI와 거리 사이의 관계를 특성화하는 로그-거리 전파 모델을 사용하여 송신자와의 거리를 추정한다. 추정된 거리는 수신자와 근접 분류를 위해 미리 정의된 기준 경계(즉, 기준 거리) 사이의 거리와 비교됩니다. 추정 거리가 기준 거리보다 짧으면 수신자는 발신자가 인접해 있다고 판단하고, 그렇지 않으면 발신자가 멀리 있다고 판단한다. 그러나 실제 환경에서는 발신자와 수신자 사이의 거리가 고정되어 있어도 RSSI가 자주 변경된다. 이는 RSSI가 무선 신호의 반사 및 회절에 의한 다중 경로 페이딩과 동일한 주파수 대역을 통해 전송되는 무선 신호에 의한 간섭에 쉽게 영향을 받기 때문이다. RSSI의 변동은 정확한 근접분류를 어렵게 하므로 RSSI 기반 근접분류의 정확도를 향상시키기 위한 많은 연구가 진행되고 있다.

[0003] 이러한 문제를 해결하기 위해 대부분의 연구는 RSSI 기반의 근접성 분류에서 거리 추정의 오차를 최소화하는 데 초점을 맞춘 이력이 있다. 수학적 모델(예: 수정된 로그 거리 모델), 통계 모델(예: 이동 평균 모델), 필터링 모델(예: Kalman 필터 모델) 및 기계 학습 모델(예: 선형 회귀 모델)은 추정 오류를 최소화하기 위해 사용되었다. 그러나 정교한 거리 추정 모델도 추정 오차가 있기 때문에 RSSI 기반의 근접성 분류의 정확성을 보장하기에는 거리 추정 모델의 개선만으로는 충분하지 않을 수 있다. 특히 송신자와 수신자의 실제 거리가 기준 경계에 가까운 경우 실제 거리가 고정되어 있어도 RSSI 기반의 근접성 분류 결과가 자주 바뀔 수 있다. 이 문제를 극복하기 위해 Nishida, D et al., Yin, L et al. 등은 참조 경계를 둘러싸는 보류 구간을 사용했다. Pending zone(보류 구간)을 사용하면 송신자와 수신자의 실제 거리가 기준 경계에 가까울 때 Pending zone 내에서 새로운 거리 추정이 아닌 이전에 추정된 거리를 유지하기 때문에 근접 분류 결과의 빈번한 변화를 완화할 수 있다. 그러나 보류 구간을 이용한 접근 방식은 주변 환경에 따라 달라지는 추정 오차에도 불구하고 Pending Zone에 대해 고정된 크기를 사용하기 때문에 정확도가 낮을 수 있다.

발명의 내용

해결하려는 과제

[0004] 본 개시에서는 RSSI 기반의 Q-Learning 기반 Pending Zone 조정을 수행하여 근접 여부를 판단하는 Tag Receiver를 제공한다.

[0005] 본 개시의 목적들은 이상에서 언급한 목적으로 제한되지 않으며, 언급되지 않은 본 개시의 다른 목적 및 장점들은 하기의 설명에 의해서 이해될 수 있고, 본 개시의 실시 예에 의해 보다 분명하게 이해될 것이다. 또한, 본 개시의 목적 및 장점들은 특허 청구 범위에 나타난 수단 및 그 조합에 의해 실현될 수 있음을 쉽게 알 수 있을 것이다.

과제의 해결 수단

[0006] 본 개시의 일 실시 예에 따른 Tag Receiver의 제어 방법은, 상기 Tag Receiver가, Tag로부터 식별 정보를 포함하는 제1 신호를 수신하는 단계, 상기 Tag Receiver가, 상기 수신된 제1 신호를 기반으로 상기 Tag에 대한 제1 거리를 추정하는 단계, 상기 Tag Receiver가, 일정한 지점에 위치하는 Anchor Beacon으로부터 주기적으로 제2 신호를 수신하는 단계, 상기 Tag Receiver가, 상기 수신된 상기 제2 신호를 바탕으로 상기 Anchor Beacon에 대한 제2 거리를 추정하는 단계, 상기 Tag Receiver가, 상기 Tag Receiver와 상기 Anchor Beacon 간의 실제 거리 및 상기 추정된 제2 거리를 기초로 노이즈 레벨을 획득하는 단계, 상기 Tag Receiver가, 상기 노이즈 레벨을 기초로 근거리 경계 및 원거리 경계를 설정하는 단계, 상기 Tag Receiver가, 상기 근거리 경계 및 상기 원거리 경계 중 적어도 하나를 상기 추정된 제1 거리와 비교하여, 상기 Tag의 사용자의 상기 Tag Receiver에 대한 근접 여부를 판단하는 단계를 포함한다.

[0007] 상기 제2 거리를 추정하는 단계는, 상기 Anchor Beacon으로부터 상기 Tag Receiver로 전송된 상기 제2 신호의

RSSI 값에 대응되는, 상기 Tag Receiver와 상기 Anchor Beacon 간의 추정 거리를 식별할 수 있다.

[0008] 이때, 상기 Tag Receiver는, 상기 근거리 경계의 상태 별로 다음 액션을 선택하기 위한 Q-value가 저장된 제1 Q-table, 상기 원거리 경계의 상태 별로 다음 액션을 선택하기 위한 Q-value가 저장된 제2 Q-table을 포함할 수 있다. 여기서, 상기 근거리 경계 및 원거리 경계를 설정하는 단계는, Q-러닝을 기반으로 하는 제1 에이전트에 상기 노이즈 레벨을 입력하여 Q-value를 획득하고, 상기 획득된 Q-value를 상기 제1 Q-table에 적용하여 상기 근거리 경계의 다음 상태를 판단하고, Q-러닝을 기반으로 하는 제2 에이전트에 상기 노이즈 레벨을 입력하여 Q-value를 획득하고, 상기 획득된 Q-value를 상기 제2 Q-table에 적용하여 원거리 경계의 다음 상태를 판단할 수 있다.

[0009] 또한, 상기 Tag Receiver의 제어 방법은, 이하 수학식에 따라 노이즈 레벨에 대한 데이터 세트를 정의하는 단계, $NL = \{\sigma_1, \sigma_2, \dots, \sigma_i\}$, 이하 수학식에 따라 상기 NL에 대한 밀도 함수를 획득하는 단계, $f(x) = \frac{1}{\sqrt{h}} \sum_{i=1}^I K\left(\frac{x - \sigma_i}{h}\right)$, 를 포함할 수 있다. 여기서 h 는 KDE의 대역폭이며, $K(x)$ 는 $K(x) = e^{-x^2/2}$ 로 주어지는 가우시안 커널 함수이다. 또한, 상기 Tag Receiver의 제어 방법은, 상기 밀도 함수의 로컬 최대값의 수에 따라 클러스터의 수를 결정하는 단계, 상기 클러스터의 수를 바탕으로 노이즈 레벨을 통해 선택되는 상태의 수를 결정하는 단계를 포함할 수 있다.

[0010] 한편, 상기 근거리 경계 및 원거리 경계를 설정하는 단계는, 이하 수학식에 따라 상기 제1 에이전트에 의해 선택된 액션을 식별할 수 있고, $a_t^{nb} = \arg \max_a Q(s_t^{nb}, a)$, 이하 수학식에 따라 상기 근거리 경계에 대한 다음 상태를 결정할 수 있으며, $s_{t+1}^{nb} = \begin{cases} s_t^{nb} - 1 & a_t^{nb} = 1, s_t^{nb} > 1 \\ s_t^{nb} + 1 & a_t^{nb} = 3, s_t^{nb} < m \\ s_t^{nb} & \text{otherwise} \end{cases}$, 이하 수학식에 따라 상기 제2 에이전트에 의해 선택된 액션을 식별하

고, $a_t^{fb} = \arg \max_a Q(s_t^{fb}, a)$, 이하 수학식 $s_{t+1}^{fb} = \begin{cases} s_t^{fb} - 1 & a_t^{fb} = 1, s_t^{fb} > 1 \\ s_t^{fb} + 1 & a_t^{fb} = 3, s_t^{fb} < m \\ s_t^{fb} & \text{otherwise} \end{cases}$ 에 따라 상기 원거리 경계에 대한 다음 상태를 결정할 수 있다.

[0011] 이 경우, 상기 근거리 경계 및 원거리 경계를 설정하는 단계는, 상기 근거리 경계 및 상기 원거리 경계 각각에 대하여 결정된 다음 상태를 바탕으로, 상기 근거리 경계 및 상기 원거리 경계 각각이 조정된 값인 d'_{nb} 및 d'_{fb} 를 획득할 수 있다. 그리고, 상기 Tag Receiver의 제어 방법은, 이하 수학식에 따라 노이즈 레벨(σ) 및 상기 조정된 값을 비교하여 상기 제1 에이전트 및 상기 제2 에이전트 각각과 관련된 보상을 산출하는 단계를 포함할 수

있고, $r_t^{nb} = \begin{cases} -1 & |d_{ref} - d'_{nb}| < \sigma/k \\ 0 & |d_{ref} - d'_{nb}| = \sigma/k \\ +1 & |d_{ref} - d'_{nb}| > \sigma/k \end{cases}$, $r_t^{fb} = \begin{cases} +1 & |d'_{fb} - d_{ref}| < \sigma/k \\ 0 & |d'_{fb} - d_{ref}| = \sigma/k \\ -1 & |d'_{fb} - d_{ref}| > \sigma/k \end{cases}$, 여기서 K 는 노이즈 레벨의 계수이고, d_{ref} 는 기준 거리이다.

[0012] 상기 제1 거리를 추정하는 단계 및 상기 제2 거리를 추정하는 단계 중 적어도 하나는, RSSI 값을 바탕으로 거리 추정을 수행하기 위한 DNN(Deep Neural Networks) 모델로, 상기 Tag로부터 순차적으로 수신되는 복수의 신호 각각의 RSSI 값을 입력할 수 있다. 상기 DNN 모델은, RSSI 값을 다중으로 입력 받기 위한 복수의 노드를 포함하는 입력 레이어, 상기 입력 레이어의 출력과 연결되는 복수의 히든 레이어, 상기 복수의 히든 레이어와 연결되고, 추정 거리를 출력하기 위한 하나의 노드를 포함하는 출력 레이어를 포함할 수 있다.

[0013] 본 개시의 일 실시 예에 따라 RSSI 값을 바탕으로 거리 추정을 수행하기 위한 DNN(Deep Neural Networks) 모델을 포함하는, Tag Receiver의 제어 방법은, 상기 Tag Receiver가, 상기 Tag로부터 순차적으로 수신되는 복수의 제1 신호 각각의 RSSI 값을 상기 DNN 모델로 입력하는 단계, 상기 Tag Receiver가, 상기 DNN 모델의 출력을 바탕으로 상기 Tag에 대한 제1 거리를 추정하는 단계, 상기 Tag Receiver가, 일정한 지점에 위치하는 Anchor Beacon으로부터 주기적으로 제2 신호를 수신하는 단계, 상기 Tag Receiver가, 상기 수신된 상기 제2 신호를 바탕으로 상기 Anchor Beacon에 대한 제2 거리를 추정하는 단계, 상기 Tag Receiver가, 상기 Tag Receiver와 상기 Anchor Beacon 간의 실제 거리 및 상기 추정된 제2 거리를 기초로 노이즈 레벨을 획득하는 단계, 상기 Tag Receiver가, 상기 노이즈 레벨을 기초로 근거리 경계 및 원거리 경계를 설정하는 단계, 상기 Tag Receiver가, 상기 근거리 경계 및 상기 원거리 경계 중 적어도 하나를 상기 추정된 제1 거리와 비교하여, 상기 Tag의 사용자 의 상기 Tag Receiver에 대한 근접 여부를 판단하는 단계를 포함한다.

발명의 효과

[0014] 본 개시의 Tag Receiver는 QPZA를 바탕으로 기존 접근 방식에 비해 더욱 높은 정확도를 가지는 것으로 확인된다.

도면의 간단한 설명

[0015] 도 1a는 본 개시의 일 실시 예에 따른 Tag Receiver의 구성을 설명하기 위한 블록도,
 도 1b는 본 개시의 일 실시 예에 따른 시스템의 아키텍처를 도시한 도면,
 도 2는 본 개시의 일 실시 예에 따른 시스템이 거리 추정을 위해 사용하는 DNN 모델의 형태를 도시한 도면,
 도 3은 본 개시의 일 실시 예에 따른 시스템의 기능 모듈을 설명하기 위한 블록 다이어그램,
 도 4a 내지 도 4b는 각각 작은 방과 큰 방 내에서 RSSI와 $\log(d)$ 간의 관계를 보여주는 선형 회귀 모델,
 도 5a 내지 도 5b는 작은 방과 큰 방 각각에 대하여 거리 추정 모델들이 RSSI 값에 대하여 추정하는 거리를 도시한 그래프들, 그리고
 도 6은 서로 다른 거리 추정 모델을 통해 획득된 작은 방과 큰 방에 대한 근접 분류의 정확도를 도시한 그래프들이다.

발명을 실시하기 위한 구체적인 내용

[0016] 본 개시에 대하여 구체적으로 설명하기에 앞서, 본 명세서 및 도면의 기재 방법에 대하여 설명한다.

[0017] 먼저, 본 명세서 및 청구범위에서 사용되는 용어는 본 개시의 다양한 실시 예들에서의 기능을 고려하여 일반적인 용어들을 선택하였다. 하지만, 이러한 용어들은 당해 기술 분야에 종사하는 기술자의 의도나 법률적 또는 기술적 해석 및 새로운 기술의 출현 등에 따라 달라질 수 있다. 또한, 일부 용어는 출원인이 임의로 선정한 용어도 있다. 이러한 용어에 대해서는 본 명세서에서 정의된 의미로 해석될 수 있으며, 구체적인 용어 정의가 없으면 본 명세서의 전반적인 내용 및 당해 기술 분야의 통상적인 기술 상식을 토대로 해석될 수도 있다.

[0018] 또한, 본 명세서에 첨부된 각 도면에 기재된 동일한 참조번호 또는 부호는 실질적으로 동일한 기능을 수행하는 부품 또는 구성요소를 나타낸다. 설명 및 이해의 편의를 위해서 서로 다른 실시 예들에서도 동일한 참조번호 또는 부호를 사용하여 설명한다. 즉, 복수의 도면에서 동일한 참조 번호를 가지는 구성요소를 모두 도시되어 있다고 하더라도, 복수의 도면들이 하나의 실시 예를 의미하는 것은 아니다.

[0019] 또한, 본 명세서 및 청구범위에서는 구성요소들 간의 구별을 위하여 "제1", "제2" 등과 같이 서수를 포함하는 용어가 사용될 수 있다. 이러한 서수는 동일 또는 유사한 구성요소들을 서로 구별하기 위하여 사용하는 것이며 이러한 서수 사용으로 인하여 용어의 의미가 한정 해석되어서는 안 된다. 일 예로, 이러한 서수와 결합된 구성요소는 그 숫자에 의해 사용 순서나 배치 순서 등이 제한되어서는 안 된다. 필요에 따라서는, 각 서수들은 서로 교체되어 사용될 수도 있다.

[0020] 본 명세서에서 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 출원에서, "포함하다" 또는 "구성되다" 등의 용어는 명세서상에 기재된 특징, 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0021] 본 개시의 실시 예에서 "모듈", "유닛", "부(part)" 등과 같은 용어는 적어도 하나의 기능이나 동작을 수행하는 구성요소를 지칭하기 위한 용어이며, 이러한 구성요소는 하드웨어 또는 소프트웨어로 구현되거나 하드웨어 및 소프트웨어의 결합으로 구현될 수 있다. 또한, 복수의 "모듈", "유닛", "부(part)" 등은 각각이 개별적인 특정한 하드웨어로 구현될 필요가 있는 경우를 제외하고는, 적어도 하나의 모듈이나 칩으로 일체화되어 적어도 하나의 프로세서로 구현될 수 있다.

[0022] 또한, 본 개시의 실시 예에서, 어떤 부분이 다른 부분과 연결되어 있다고 할 때, 이는 직접적인 연결뿐 아니라, 다른 매체를 통한 간접적인 연결의 경우도 포함한다. 또한, 어떤 부분이 어떤 구성요소를 포함한다는 의미는, 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라 다른 구성요소를 더 포함할 수 있는 것을 의미한다.

- [0023] 이하에서, 첨부된 도면을 이용하여 본 발명의 다양한 실시 예들에 대하여 구체적으로 설명한다.
- [0024] 본 개시에서는 RSSI 기반의 Proximity Classification를 위한 Q-Learning 기반 Pending Zone 조정, 즉 QPZA가 적용된 시스템을 제안한다. QPZA는 주변 환경의 변화를 고려하여 보류 구간의 크기를 적응적으로 조정하여 정확도를 향상시키기 위해 Tag Receiver(ex. BLE Tag Receiver)에서 실행될 수 있다. 특히 QPZA는 Q-러닝을 사용하여 노이즈 레벨이 증가하면 보류 구간의 크기를 확장하고 그렇지 않으면 감소시킨다. Q-러닝은 Q-러닝 에이전트가 누적 보상을 고려하여 다음 상태에 대한 최선의 조치를 찾는 모델 없는 강화 학습 기법이다. QPZA에서 작업은 보류 구간의 크기 조정을 수반하며 보상은 현재 노이즈 수준에 대한 작업의 적합성을 나타낸다. 상태(state)는 수신기와 보류 구간의 경계 사이의 거리를 나타낸다. 보류 구간의 크기는 두 결정 경계 사이의 영역, 즉 가까운 경계와 먼 경계 사이의 영역이다. 따라서 QPZA는 두 개의 Q-러닝 에이전트를 사용하여 근거리 경계와 원거리 경계를 별도로 조정한다. 각 Q-learning 에이전트의 상태 수를 결정하기 위해 평균 이동 클러스터링 알고리즘(mean-shift clustering algorithm)을 사용하여 노이즈 레벨 데이터 세트를 클러스터링하고 각 클러스터의 중심을 고려하여 각 상태의 값을 계산한다. QPZA는 두 가지 보상 기능을 사용하여 가까운 경계와 먼 경계에 대한 보상을 개별적으로 설정할 수 있다.
- [0025] 도 1a는 본 개시의 일 실시 예에 따른 Tag Receiver의 구성을 설명하기 위한 블록도이다.
- [0026] Tag Receiver(100)는 사용자 등이 소지한 Tag를 인식하여 Tag의 근접 여부를 판단하기 위한 전자 기기 내지 단말 기기에 해당한다.
- [0027] 도 1a를 참조하면, Tag Receiver(100)는 메모리(110), 통신부(120), 및 프로세서(130)를 포함할 수 있다.
- [0028] 메모리(110)는 Tag Receiver(100)의 구성요소들의 전반적인 동작을 제어하기 위한 운영체제(OS: Operating System) 및 Tag Receiver(100)의 구성요소와 관련된 적어도 하나의 인스트럭션 또는 데이터를 저장하기 위한 구성이다.
- [0029] 메모리(110)는 ROM, 플래시 메모리 등의 비휘발성 메모리를 포함할 수 있으며, DRAM 등으로 구성된 휘발성 메모리를 포함할 수 있다. 또한, 메모리(110)는 하드 디스크, SSD(Solid state drive) 등을 포함할 수도 있다.
- [0030] 통신부(120)는 다양한 유무선 통신방식으로 적어도 하나의 외부 장치와 통신을 수행하기 위한 회로, 모듈, 칩 등을 포함할 수 있다.
- [0031] 통신부(120)는 다양한 네트워크를 통해 외부의 칩 내지는 장치들(ex. Tag, Anchor Beacon, 서버 등)과 연결될 수 있다.
- [0032] 구체적인 예로, 통신부(120)는 블루투스, BLE, NFC, RFID, WiFi Direct 등 다양한 근접 통신 방식을 바탕으로 Tag 및/또는 Anchor Beacon과 통신을 수행할 수 있다. 이때, Tag Receiver(100)는 수신되는 신호의 세기, 신호의 송수신 시간 등을 바탕으로 거리를 추정할 수 있다.
- [0033] 또한, Tag Receiver(100)는 유선 통신 방식 및/또는 무선 통신 방식을 통해 서버와 연결될 수 있다. 구체적으로, 하나 이상의 네트워크를 통해 Tag Receiver(100)와 서버 간의 통신이 수행될 수 있다. 네트워크는 영역 또는 규모에 따라 개인 통신망(PAN; Personal Area Network), 근거리 통신망(LAN; Local Area Network), 광역 통신망(WAN; Wide Area Network) 등일 수 있으며, 네트워크의 개방성에 따라 인트라넷(Intranet), 엑스트라넷(Extranet), 또는 인터넷(Internet) 등일 수 있다. 이때, LTE(long-term evolution), LTE-A(LTE Advance), 5G(5th Generation) 이동통신, CDMA(code division multiple access), WCDMA(wideband CDMA), UMTS(universal mobile telecommunications system), WiBro(Wireless Broadband), GSM(Global System for Mobile Communications), TDMA(Time Division Multiple Access), WiFi(Wi-Fi), WiFi Direct, Bluetooth, NFC(near field communication), Zigbee 등 다양한 무선 통신 방식이 활용될 수 있다. 또한, 이더넷(Ethernet), 광 네트워크(optical network), USB(Universal Serial Bus), 선더볼트(ThunderBolt) 등의 유선 통신 방식이 활용될 수도 있다.
- [0034] 프로세서(130)는 Tag Receiver(100)를 전반적으로 제어하기 위한 구성이다. 구체적으로, 프로세서(130)는 메모리(110)와 연결되는 한편 메모리(110)에 저장된 적어도 하나의 인스트럭션을 실행함으로써 본 개시의 다양한 실시 예들에 따른 동작을 수행할 수 있다.
- [0035] 프로세서(130)는 CPU, AP, DSP(Digital Signal Processor) 등과 같은 범용 프로세서, GPU, VPU(Vision Processing Unit) 등과 같은 그래픽 전용 프로세서 또는 NPU와 같은 인공지능 전용 프로세서 등을 포함할 수 있다. 인공지능 전용 프로세서는, 특정 인공지능 모델의 훈련 내지는 이용에 특화된 하드웨어 구조로 설계될 수

있다.

- [0036] 도 1b는 QPZA를 사용한 RSSI 기반 근접 분류 시스템의 아키텍처를 보여준다. 도 1에서 시스템은 Tag, Anchor Beacon, 및 Tag Receiver로 구성된다. 여기서, 상술한 장치들은 BLE, 적외선 통신, NFC, RFID 등 다양한 근접 통신 방식을 활용할 수 있으며, 바람직하게는 BLE 통신이 활용될 수 있다.
- [0037] BLE 태그는 UUID(Universal Unique Identifier), major ID, minor ID 및 전송 전력(TxPower)을 포함한 비콘 정보를 BLE Tag Receiver에게 알리기 위해 게시 패킷을 브로드캐스팅하는 역할을 한다. UUID, major ID, minor ID는 BLE 태그를 식별하는 데 사용되며, TxPower는 식별된 BLE 태그와 BLE Tag Receiver 사이의 거리를 추정하는 데 사용된다.
- [0038] Anchor Beacon은 특정 위치에 배치되는 특수 유형의 BLE 태그이다. Anchor Beacon은 BLE 태그와 동일한 방식으로 게시 패킷을 브로드캐스트한다. Anchor Beacon의 비콘 정보는 주변 환경의 노이즈 레벨을 측정하기 위한 용도로만 사용될 수 있다.
- [0039] BLE Tag Receiver는 게시 패킷을 수신할 때마다 RSSI를 측정하고 측정된 RSSI를 이용하여 거리 추정을 수행할 수 있다. RSSI 기반 근접 분류의 정확도에 대한 보류 구간 조정의 효과를 철저히 조사하기 위해 거리 추정이 측정된 RSSI의 원시 값을 기반으로 한다고 가정한다. BLE 태그로부터 패킷을 수신하면 BLE Tag Receiver는 근접성 분류를 수행하여 BLE 태그가 근처에 있는지 여부를 판단한다. 다른 경우에는 노이즈 레벨을 계산하고 QPZA를 수행하여 보류 구간의 크기를 조정할 수 있다.
- [0040] 도 1b에서 보류 구간(Pending Zone)은 가까운 경계(: 근거리 경계, Near Boundary)와 먼 경계(: 원거리 경계, Far Boundary) 사이의 영역으로 표시된다. 보류 구간에는 시스템에서 미리 정의한 기준 경계가 포함된다. 따라서 BLE Tag Receiver와 가까운 경계선(d_{nb}) 사이의 거리가 BLE Tag Receiver와 기준 경계선(d_{ref}) 사이의 거리보다 작거나 같고, BLE Tag Receiver와 먼 경계선(d_{nf}) 사이의 거리가 기준 경계선보다 크거나 같을 수 있다.
- [0041] 근접성 분류를 위해, BLE Tag Receiver는 추정 거리가 d_{nb} 보다 작은 경우 BLE 태그가 근접해 있는(예를 들어, true) 것으로 판단할 수 있다. 반대로, BLE Tag Receiver는 추정 거리가 d_{nf} 보다 작은 경우 BLE 태그가 근접해 있지 않은(예를 들어, false) 것으로 판단할 수 있다.
- [0042] 만약, BLE 태그의 추정 거리가 보류 구간 내에 있으면 BLE Tag Receiver는 이전 결과와 동일한 근접성 분류 결과를 유지할 수 있다(즉, 결정 보류). 이러한 맥락에서 d_{nb} 및 d_{nf} 는 근접 분류의 핵심 기준이다.
- [0043] 그리고, 본 개시의 QPZA는 노이즈 레벨을 고려하여 적응적으로 조정하여 근접 분류의 정확도를 향상시킨다.
- [0044] 게시 패킷을 수신한 BLE Tag Receiver는 RSSI를 측정하고 거리 추정 모델을 사용하여 거리를 추정한다. 이를 위해 기존 거리 추정 모델의 세 가지 유형인 iBeacon, 선형 회귀, 및 딥 러닝 모델이 고려될 수 있다. 그런 다음 정확도가 가장 높은 모델 중 하나를 선택하여 BLE Tag Receiver에 적용할 수 있다.
- [0045] iBeacon 모델은 이하 수학적 식 1의 iBeacon 사양에 의해 주어질 수 있다.

수학적 식 1

[0046]
$$ed_{iBeacon} = 0.89976 \left(\frac{RSSI}{TxPower} \right)^{2.7088} + 0.111$$

- [0047] 이때, $ed_{iBeacon}$ 는 iBeacon 모델의 추정 거리이며, RSSI는 측정된 RSSI의 값이다. RSSI의 단위는 dBm에 해당한다. 상기 수학적 식에서, $ed_{iBeacon}$ 은 RSSI에 비례하며 TxPower에는 반비례한다. iBeacon 모델은 특정한 환경에서 수집된 RSSI 데이터셋을 바탕으로 구축된 것이다. 그러므로, 이는 환경의 변화와 무관하게 고정된 상태로 유지된다.
- [0048] 선형 회귀 모델을 구축하기 위해 RSSI와 log-distance 사이의 관계는 수학적 식 2로 주어진다고 가정한다.

수학식 2

$$\log(d) = A(RSSI) + B$$

여기서, d 는 BLE Tag Receiver와 BLE Tag 간의 거리이며, A 및 B 는 각각 계수(즉, 회귀 기울기) 및 $\log(d)$ 의 절편에 해당한다. D 의 단위는 미터이다. 수학식 2 내의 A 및 B 를 결정하기 위해, RSSI 기반의 거리 추정을 위한 로그-거리 데이터셋(LD)이 이하 수학식 3과 같이 정의되어 활용될 수 있다.

수학식 3

$$LD = [(RSSI_0, \log(d_0)), (RSSI_1, \log(d_1)), \dots, (RSSI_n, \log(d_n))]$$

$RSSI_n$ 은 $(n+1)$ 번째 advertising 패킷에 대해 측정된 RSSI의 값에 해당한다. d_n 은 $(n+1)$ 번째 advertising 패킷에 대한 실제 거리에 해당한다. 또한, 일반 최소 제곱은 매개변수 A 및 B 를 계산하는 데 사용된다. 구체적으로, 파라미터 A 및 B 는 이하 수학식 4 및 5에 따라 각각 획득될 수 있다.

수학식 4

$$A = \frac{\sum_{i=0}^n (RSSI_i - \text{mean}(RSSI))(\log(d_i) - \text{mean}(\log(d)))}{\sum_{i=0}^n (RSSI_i - \text{mean}(RSSI))^2}$$

수학식 5

$$B = \text{mean}(\log(d)) - (\text{mean}(RSSI) \times A)$$

이때, $\text{mean}(RSSI)$ 는 LD 내 각각의 RSSI의 평균이며, $\text{mean}(\log(d))$ 는 LD 내 각각의 $\log(d)$ 의 평균이다. A 및 B 파라미터를 획득함에 있어서, 수학식 6으로 표현된 거리 추정 모델이 BLE tag receiver 내에 구비될 수 있다. 즉, BLE tag receiver가 advertisement 패킷을 수신할 때마다, BLE tag receiver는 거리 추정 모델을 이용하여 거리를 추정할 수 있다.

수학식 6

$$ed_{LR} = 10^{(A(RSSI_{\text{mean}}) + B)}$$

여기서, $RSSI_{\text{mean}}$ 이 측정된 RSSI 값이라고 할 때, ed_{LR} 은 선형 회귀 모델의 추정 거리이다. 선형 회귀 모델은 RSSI 값에 의존하여 변하기 때문에, RSSI 수집을 위한 환경이 변하면 선형 회귀 모델 역시 변경될 수 있다.

딥 러닝 모델을 구축하기 위해 먼저 여러 RSSI 입력을 사용하여 하나의 추정 거리(즉, ed_{DL})를 도출하는 다대일 DNN(Deep Neural Networks) 모델이 설계될 수 있다. 도 2는 다대일 DNN 모델의 형태를 보여주며, 본 DNN 모델은 하나의 입력 레이어, 히든 레이어들, 그리고 하나의 출력 레이어로 구성된다.

입력 레이어는 순차적으로 생성되는 다중 RSSI 입력을 입력하기 위한 다중 노드(즉, 그림의 원)로 구성된다. 도 2에서 RSSI 입력의 수는 1에 해당한다. 히든 레이어의 수와 각 히든 레이어의 노드 수는 높은 정확도를 얻기 위해 경험적으로 결정된다. 또한 역전파(back-propagation) 과정에서 그래디언트가 점차 0으로 수렴하는 그래디언트 소실(loss) 문제를 해결하기 위한 활성화 함수로 ReLU(rectified linear unit)가 사용될 수 있다. 선형 회

귀 모델과 유사하게 딥러닝 모델은 RSSI 수집 환경에 따라 달라진다. 그러나 다른 모델과 달리 거리를 추정하기 위해서는 여러 개의 RSSI 입력이 필요하다.

- [0060] 본 개시에 따른 시스템은, 주변 환경의 노이즈 수준에 따라 달라지는 근접 분류의 정확도를 향상시키기 위해 Pending Zone의 크기를 조정하도록 설계될 수 있다.
- [0061] 시스템의 QPZA에서 BLE Tag Receiver는 먼저 Anchor Beacon을 사용하여 노이즈 레벨을 확인한 다음 노이즈 레벨에 따라 Pending Zone의 크기를 조정할 수 있다. QPZA는 노이즈 레벨이 증가하면 보류 구간의 크기를 확장한다. 이 경우 d_{nb} 와 d_{fb} 의 차이가 커진다. 반대의 경우 d_{nb} 와 d_{fb} 의 차이는 감소한다.
- [0062] 한편, 일 실시 예에 따르면, 노이즈 레벨에 따라 보류 구간의 크기가 일정 면적 이상이 되는 경우 Tag Receiver는 Tag에 대한 근접 감지를 중단할 수 있다.
- [0063] 이때, Tag Receiver는 보류 구간의 크기가 일정 면적 이상이 된 시점의 노이즈 레벨을 대상 노이즈 레벨로 기록할 수 있다.
- [0064] 또한, Tag Receiver는 일정 주기에 따라 획득되는 노이즈 레벨을 바탕으로 예측 모델(ex. RNN 모델)을 학습시켜, 다가올 하나 이상의 주기의 노이즈 레벨을 예측할 수 있다. 구체적으로, 매 주기마다, Tag Receiver는 Anchor Beacon으로부터 수신되는 신호에 따라 Anchor Beacon에 대한 거리를 추정하고, 추정된 거리를 실제 거리와 비교하여 그 차이가 클수록 노이즈 레벨을 높게 식별할 수 있다.
- [0065] 예를 들어, 현 시점을 기준으로 과거의 노이즈 레벨이 순차적으로 예측 모델에 입력된 결과, 다음 주기의 노이즈 레벨이 대상 노이즈 레벨 이상인 것으로 예측될 수 있다. 이 경우, Tag Receiver는 다음 주기에 대해서는 유희 주기로 설정하고, Tag에 대한 거리 추정 및 Anchor Beacon에 대한 거리 추정을 수행하지 않을 수 있다. 그 결과, 유희 주기에 대해서는 근접 여부에 대한 판단 역시 수행되지 않을 수 있다.
- [0066] 여기서, Tag Receiver는 과거 주기의 노이즈 레벨 및 유희 주기에 대해 예측된 상기 노이즈 레벨을 바탕으로, 예측 모델을 통해, 유희 주기 다음에 해당하는 주기의 노이즈 레벨을 예측할 수 있다. 다만, 만약 유희 주기 다음에 해당하는 주기에 대해 측정된 노이즈 레벨이 예측된 레벨을 기준으로 일정 범위를 벗어나는 경우, Tag Receiver는 대상 노이즈 레벨을 상향하여 업데이트할 수 있다. 즉, 예측 모델의 정확도가 높지 않음에 따라, 유희 주기가 발생하기 위한 조건이 어려워질 수 있다. 반면, 유희 주기 다음에 해당하는 주기에 대해 측정된 노이즈 레벨이 예측된 레벨을 기준으로 일정 범위 내인 경우, 상술한 대상 노이즈 레벨이 유지될 수 있다.
- [0067] 이렇듯 유희 주기가 운영되는 경우, 정확도가 낮은 환경에서 불필요한 거리 추정 내지 근접 감지에 로드가 낭비되는 상황이 방지될 수 있다.
- [0068] 도 3은 QPZA의 작동 블록 다이어그램을 보여준다. 도 3에서 QPZA는 노이즈 레벨 계산기(Noise Level Calculator), 근거리 경계 조정기(Near Boundary Adjuster) 및 원거리 경계 조정기(Far Boundary Adjuster)로 구성된다. 본 구성들은 소프트웨어 및/또는 하드웨어로 구현되는 기능적 블록들로, Tag Receiver 내에 포함될 수 있다.
- [0069] 또는, 상기 기능적 블록들이 Tag Receiver와 통신이 가능한 적어도 하나의 서버 상에 포함될 수도 있다. 구체적으로, 신호의 RSSI에 따른 거리 추정 기능, 노이즈 레벨에 따른 경계 설정 기능 중 적어도 하나가 서버 상에서 수행될 수도 있다.
- [0070] 노이즈 레벨 계산기는 노이즈 레벨(σ)을 계산하는 데 사용되며 근거리 및 원거리 경계 조정기는 각각 d_{nb} 및 d_{fb} 를 조정하는 데 사용된다. 각 경계의 조정기는 Q-러닝 에이전트(Q-learning Agent)와 보상 계산기(Reward Calculator)를 포함할 수 있다. 전자는 다음 d_{nb} 또는 d_{fb} (즉, d'_{nb} 또는 d'_{fb})를 결정하기 위해 동작(즉, 감소, 유지 및 증가)을 선택한다. 후자는 선택한 동작이 노이즈 수준에 적합한지 여부를 확인한 다음 확인 결과를 고려하여 보상을 계산한다. 보상은 Q-learning 에이전트를 업데이트하는 데 사용된다.
- [0071] BLE Tag Receiver는 Anchor Beacon에서 advertising 패킷을 수신할 때마다 QPZA를 수행할 수 있다. 구체적으로 Anchor Beacon의 추정 거리(즉, ed_{anc})를 얻은 후 노이즈 수준 계산기를 실행하여 σ 를 결정할 수 있다. 이는, ed_{anc} 와, BLE Tag Receiver 및 Anchor Beacon 간의 거리(즉, d_{anc})의 차이를 바탕으로 산출된다. 따라서 QPZA에서 노이즈 레벨은 미터로 표시될 수 있다. 그런 다음 근거리 및 원거리 경계 조정기는 누적 보상을 고려하여 d_{nb} 및 d_{fb} 를 조정할 수 있다. 이를 위해 조정자별 Q-learning 에이전트는 Q-table을 유지한다. 표 1과 2는 각각 가

가운 경계에 대한 Q-table($\mathbf{Q}$)과 먼 경계에 대한 Q-table($\hat{\mathbf{Q}}$)을 보여준다.

표 1

Q-Table for Near		Action		
Boundary (Q)		Decrease	Keep	Increase
State	nb_1	$Q(1,1)$	$Q(1,2)$	$Q(1,3)$
	nb_2	$Q(2,1)$	$Q(2,2)$	$Q(2,3)$
	...	...	...	...
	nb_m	$Q(m,1)$	$Q(m,2)$	$Q(m,3)$

표 2

Q-Table for Far		Action		
Boundary ($\hat{Q}$)		Decrease	Keep	Increase
State	fb_1	$\hat{Q}(1,1)$	$\hat{Q}(1,2)$	$\hat{Q}(1,3)$
	fb_2	$\hat{Q}(2,1)$	$\hat{Q}(2,2)$	$\hat{Q}(2,3)$
	...	...	...	...
	fb_m	$\hat{Q}(m,1)$	$\hat{Q}(m,2)$	$\hat{Q}(m,3)$

Q 테이블에서 각 행은 각 상태에 대한 후보 거리를 나타내며 BLE Tag Receiver와 경계 사이의 거리로 선택될 수 있다. 상태는 후보 거리의 인덱스이며 상태의 수는 두 경계에 대해 동일하다. 따라서 d_{nb} , d'_{nb} , d_{fb} , d'_{fb} 는 각각 수학적 7과 8로 주어질 수 있다.

수학적 7

$$d_{nb}, d'_{nb} \in \{nb_1, nb_2, \dots, nb_m\}$$

수학적 8

$$d_{fb}, d'_{fb} \in \{fb_1, fb_2, \dots, fb_m\}$$

이때, nb_m 은 가까운 경계에 대한 Q 테이블 내의 m번째 후보 거리이고, fb_m 은 먼 경계에 대한 Q 테이블 내의 m번째 후보 거리이다. Q-table의 열은 감소, 유지, 증가를 포함한 조치를 나타내며 각 조치에 대한 인덱스는 각각 1, 2, 및 3이다. Q-learning 에이전트가 액션을 감소 또는 증가로 결정하고 변경하면, d_{nb} 및 d_{fb} 는 변경된다. 반면, 그렇지 않으면 이전 거리로 유지된다. Q 테이블의 요소들은 상태-액션 값 또는 Q 값을 나타낸다. Q 값은 근거리 및 원거리의 경계 각각에 대하여 $Q(s_{nb}, a_{nb})$ 및 $\hat{Q}(s_{fb}, a_{fb})$ 로 표현된다. 각각의 경계에 있어서, s_{nb} 및 s_{fb} 는 후보 거리의 지수(즉, 상태)를 나타내며, a_{nb} 및 a_{fb} 는 액션의 지수를 나타낸다.

상태의 수(즉, m)는 인접한 BLE 태그의 수가 다른 환경에서 수집된 노이즈 레벨 데이터 샘플로 구성된 노이즈 레벨 데이터 세트(NL)의 클러스터 수를 기반으로 결정될 수 있다. 노이즈 수준 데이터 세트는 이하 수학적 9로 제공될 수 있다.

수학식 9

[0079] $\mathbf{NL} = \{\sigma_1, \sigma_2, \dots, \sigma_l\}$

[0080] 여기서 l 은 $\mathbf{NL}$ 내 노이즈 레벨 데이터 샘플의 수이다. $\mathbf{NL}$ 에서 클러스터 수를 결정하기 위해 평균 이동 클러스터링 알고리즘이 사용될 수 있다. 본 알고리즘은 KDE(Kernel Density Estimation)를 사용하여 주어진 데이터 세트에 대한 밀도 함수를 생성하고 밀도 함수에서 로컬 최대값을 검색할 수 있다. 클러스터 수는 밀도 함수의 로컬 최대값 수와 같다. $\mathbf{NL}$ 에 대한 밀도 함수는 수학식 10과 같이 KDE를 사용하여 얻을 수 있다.

수학식 10

[0081] $f(x) = \frac{1}{ll} \sum_{i=1}^l K\left(\frac{x - \sigma_i}{h}\right)$

[0082] $f(x)$ 는 $\mathbf{NL}$ 의 밀도 함수이며, h 는 KDE의 대역폭(스케일로 지칭되기도 함)이며, $K(x)$ 는 $K(x) = e^{-x^2/2}$ 로 주어지는 가우시안 커널 함수이다. h 의 값은 클러스터 간 분산을 최대화하고 클러스터 간 분산을 최소화하도록 결정될 수 있다. $f(x)$ 내 로컬 최대값의 수는 $f'(x)=0$ 및 $f''(x)<0$ 를 만족하는 인수(i.e., x_s)를 카운팅하여 획득된다. $f'(x)$ 및 $f''(x)$ 는 각각 $f(x)$ 의 1차 미분 및 2차 미분이다. 이때, $x > 0$ 및 $x \in \mathbb{R}$ 를 만족한다. 각각의 로컬 최대값에 대한 인수의 값은 각 클러스터의 중심을 나타낸다. 마침내, m 은 로컬 최대값의 합계 및 기준 경계의 수에 따라 결정된다(즉, 1).

[0083] 각 상태에 대한 후보 거리 값은 기준 거리와 클러스터 중심을 사용하여 결정된다. 구체적으로, 상태가 증가함에 따라 가까운 경계에 대한 후보 거리는 증가하고 먼 경계에 대한 후보 거리는 감소할 수 있다. 가까운 경계와 먼 경계에 대한 i -번째 후보 거리의 값(즉, nb_i 및 fb_i)은 각각 수학식 11 및 수학식 12를 사용하여 계산된다.

수학식 11

[0084] $nb_i = d_{ref} - c_{m-i}$

수학식 12

[0085] $fb_i = d_{ref} + c_{i-1}$

[0086] c_{m-i} 는 $(m-i)$ 번째 클러스터의 중심이며, c_{i-1} 은 $(i-1)$ 번째 클러스터의 중심이다. 여기서, c_0 는 0과 같다. 그러므로, nb_m 및 fb_1 은 d_{ref} 와 같다.

[0087] d'_{nb} 및 d'_{fb} 를 결정하기 위해, 각각의 Q 러닝 에이전트는 Q 테이블 내 현재 상태에서 가장 큰 Q 값의 액션을 선택할 수 있다. 만약 둘 이상의 액션이 동일한 가장 큰 Q 값을 가지는 경우, Q 러닝 에이전트는 그들 중 하나를 임의로 선택할 수 있다. 각각의 Q 러닝 에이전트에 의해 선택된 액션은 이하 수학식 13 및 14를 통해 각각 주어질 수 있다.

수학식 13

$$a_t^{nb} = \arg \max_a Q(s_t^{nb}, a)$$

수학식 14

$$a_t^{fb} = \arg \max_a \hat{Q}(s_t^{fb}, a)$$

이때, s_t^{nb} 및 s_t^{fb} 는 각각 근거리 및 원거리의 경계들에 대한 현재 상태이고, a_t^{nb} 및 a_t^{fb} 는 각각 근거리 및 원거리의 경계들에 대해 선택된 액션의 지수이다. 이때, t는 QPZA의 실행 수와 동일한 것으로 가정되며, 두 경계의 초기 상태는(즉, s_0^{nb} 및 s_0^{fb}) $\{nb_1, nb_2, \dots, nb_m\}$ 및 $\{fb_1, fb_2, \dots, fb_m\}$ 내에서 지정된 것이다. 액션을 선택함에 있어서, 다음 상태(즉, s_{t+1}^{nb} 및 s_{t+1}^{fb})는 이하 수학식 15 및 16에 따라 결정된다.

수학식 15

$$s_{t+1}^{nb} = \begin{cases} s_t^{nb} - 1 & a_t^{nb} = 1, s_t^{nb} > 1 \\ s_t^{nb} + 1 & a_t^{nb} = 3, s_t^{nb} < m \\ s_t^{nb} & \text{otherwise} \end{cases}$$

수학식 16

$$s_{t+1}^{fb} = \begin{cases} s_t^{fb} - 1 & a_t^{fb} = 1, s_t^{fb} > 1 \\ s_t^{fb} + 1 & a_t^{fb} = 3, s_t^{fb} < m \\ s_t^{fb} & \text{otherwise} \end{cases}$$

그리고, 각각의 Q 러닝 에이전트는 d'_{nb} 및 d'_{fb} 를 각각 $nb_{s_{t+1}^{nb}}$ 및 $fb_{s_{t+1}^{fb}}$ 로 획득할 수 있다. 예를 들어, 만약 선택된 액션이 감소이고 $a_t^{nb} = nb_3$ (즉, $s_t^{nb} = 3$)인 경우, $d'_{nb} = nb_2$ (즉, $s_{t+1}^{nb} = 2$).

보류 구간의 크기를 적응적으로 조정하기 위해 QPZA를 실행할 때마다 보상을 기반으로 Q 테이블이 업데이트된다. 보상은 계산된 노이즈 수준에 대해 선택한 작업의 적합성을 나타낸다. 결과적으로 각 경계에 대한 보상의 값은 조정된 경계와 계산된 노이즈 수준의 차이에 따라 달라진다. 예를 들어, 계산된 노이즈 레벨이 조정된 Pending Zone의 크기보다 크면 가까운 경계(즉, r_t^{nb})에 대한 보상 값은 음수이고 먼 경계(즉, r_t^{fb})에 대한 보상 값은 양수이다. 따라서 대기 중인 보류 구간의 크기가 확장된다. 보상 계산기는 계산된 노이즈 수준(즉, σ)과 조정된 경계(즉, d'_{nb} 및 d'_{fb})를 비교하여 보상을 계산할 수 있다. 그러므로, r_t^{nb} 및 r_t^{fb} 는 이하 수학식 17 및 18에 의해 획득될 수 있다.

수학식 17

$$r_t^{nb} = \begin{cases} -1 & |d_{req} - d'_{nb}| < \sigma/k \\ 0 & |d_{req} - d'_{nb}| = \sigma/k \\ +1 & |d_{req} - d'_{nb}| > \sigma/k \end{cases}$$

수학식 18

$$r_i^{\text{nb}} = \begin{cases} +1 & |d_{i,\text{nb}}^{\text{nb}} - \hat{d}_{i,\text{nb}}^{\text{nb}}| < \sigma/k \\ 0 & |d_{i,\text{nb}}^{\text{nb}} - \hat{d}_{i,\text{nb}}^{\text{nb}}| = \sigma/k \\ -1 & |d_{i,\text{nb}}^{\text{nb}} - \hat{d}_{i,\text{nb}}^{\text{nb}}| > \sigma/k \end{cases}$$

여기서 k는 노이즈 레벨 계수이다.

보상 계산기가 보상을 결정하면 Q-learning 에이전트는 그에 따라 Q-table을 업데이트한다. 구체적으로 각 경계에 대한 Q-table은 각각 수학식 19와 수학식 20을 사용하여 업데이트될 수 있다.

수학식 19

$$Q(s_i^{\text{nb}}, a_i^{\text{nb}}) \leftarrow Q(s_i^{\text{nb}}, a_i^{\text{nb}}) + \alpha \times [r_i^{\text{nb}} + \gamma \times \max_a Q(s_{i+1}^{\text{nb}}, a) - Q(s_i^{\text{nb}}, a_i^{\text{nb}})]$$

수학식 20

$$Q(s_i^{\text{nb}}, a_i^{\text{nb}}) \leftarrow Q(s_i^{\text{nb}}, a_i^{\text{nb}}) + \alpha \times [r_i^{\text{nb}} + \gamma \times \max_a Q(s_{i+1}^{\text{nb}}, a) - Q(s_i^{\text{nb}}, a_i^{\text{nb}})]$$

여기서, γ 는 미래 보상의 중요도를 나타내는 discount factor이고, α 는 새로 얻은 Q-값이 이전 Q-값을 대체하는 정도를 결정하는 학습률이다. 두 매개변수 γ 및 α 모두 0 및 1의 사이에 해당하는 값으로 설정된다.

이렇듯, 본 개시의 시스템으로 구현되는 QPZA는 보류 구간의 크기를 적응적으로 조정하여 RSSI 기반 근접성 분류의 정확도를 향상시키도록 설계되었다.

본 개시의 시스템의 QPZA의 성능을 평가하기 위해, 보류 구간의 크기를 고정된 크기로 설정하는 기존 시스템의 성능과 비교했다.

일 예로, 본 개시의 QPZA에 따른 Anchor Beacon, BLE 태그 및 BLE Tag Receiver는 Quad-Core Cortex-A72 1.5GHz CPU 및 2GB RAM을 탑재하고 Raspberry Pi OS에서 작동하는 Raspberry Pi 4 Model B에서 구현되었다. BlueZ 라이브러리를 사용하여 2.4GHz ISM 대역에서 작동하는 장치의 BLE 통신 기능을 개발했다. Anchor Beacon 및 BLE 태그는 TxPower가 -59dBm으로 설정된 상태에서 500ms마다 iBeacon 패킷 형식으로 개시 패킷을 전송하도록 설정되었다. iBeacon 패킷 형식에는 UUID, major ID, minor ID 및 TxPower를 포함하는 30바이트의 고정 길이 데이터가 있다. 거리 추정 모델과 QPZA 모델은 BLE Tag Receiver에 탑재된다. 거리 추정 모델과 QPZA 개발을 위해 MySQL 기반 데이터베이스(DB) 서버를 이용하여 RSSI 데이터를 수집하였다. DB 서버의 사양은 표 3과 같다.

표 3

Component	Description
Operating system	Windows 10 Pro 64 bit
Processor	Intel core i7-8700 CPU 3.20 GHz
HDD	500 GB
RAM	8 GB
Web server	Apache Tomcat
Database	MySQL

작은 방(8 x 13 m²)과 큰 방(15 x 21 m²)의 두 가지 환경에서 RSSI 데이터가 수집되었다. 두 환경 모두 방의 중앙에 BLE Tag Receiver를 배치하고 Anchor Beacon과 BLE Tag Receiver 사이의 거리(d_{anc})는 1m로 설정되었다.

작은 방의 경우 BLE 태그와 BLE Tag Receiver 사이의 실제 거리는 0.50, 1.00, 1.50, 2.00, 2.50, 3.00, 4.00m로 설정되었다. 큰 방의 경우에는 0.50, 1.00, 1.50, 2.00, 2.50, 3.00, 4.00, 5.00, 6.00, 7.00m로 설정되었다. 수집된 데이터 세트(즉, **LD** 및 **NL**)는 7:3의 비율로 훈련 데이터 세트와 테스트 데이터 세트로 나뉜다. **LD** 및 **NL** 데이터 세트를 사용하여 선형 회귀 기반 거리 추정 모델과 오픈 소스 Python 라이브러리인 scikit-learn을 사용하여 QPZA를 구축했다.

[0108] 도 4a, 4b는 각각 작은 방과 큰 방 내에서 RSSI와 log(d) 간의 관계를 보여주는 선형 회귀 모델을 도시한 것이다. 도면에서 검은색 점은 **LD**의 훈련 데이터 세트의 데이터를 나타내고 빨간색 선은 수학적 2의 결과를 나타낸다. 두 경우 모두 BLE Tag Receiver와 BLE 태그 사이의 거리가 멀어질수록 신호 강도가 감소하는 경향이 있기 때문에 RSSI 값이 증가할수록 log(d)의 값도 감소한다. 학습된 선형 회귀 모델은 다중 경로 페이딩과 같은 환경 변화로 인해 달라질 수 있다. 구체적으로 작은 방에 대한 매개변수는 -0.02283564와 -1.35625393이고, 큰 방에 대한 매개변수는 -0.0327501과 -1.96797933이다. 결과적으로 각 방에 대한 선형 회귀 기반 거리 추정 모델을 얻었으며 이는 수학적 21과 수학적 22로 표현된다.

수학적 21

[0109]
$$ed_{sr} = 10^{(-0.02283564 \times RSSI - 1.35625393)}$$

수학적 22

[0110]
$$ed_{lr} = 10^{(-0.0327501 \times RSSI - 1.96797933)}$$

[0111] 여기서, ed_{sr} 및 ed_{lr} 은 각각 작은 방과 큰 방의 추정 거리이다.

[0112] 딥 러닝 모델을 만들기 위해 Keras 및 TensorFlow Python 오픈 소스 라이브러리가 사용되었다. 입력 레이어의 노드 수는 두 방 모두에 대해 5개로 설정되어 5개의 RSSI 입력으로 거리를 추정할 수 있게 설계되었다. 은닉층의 수는 4개로 설정되었 각 은닉층의 노드 수는 64, 64, 32, 16으로 설정되었다. batch size, epoch, 학습률(learning rate)은 각각 200, 200, 0.0001로 설정되었다.

[0113] QPZA에서 사용되는 Q-table을 생성하기 위해 훈련 데이터 세트에 대해 평균 이동 클러스터링이 수행되었다. KDE의 대역폭은 0.32로 설정되었고 각 방에 대해 1.5m와 2.5m의 기준 거리가 사용되었다. 각 방에 대해 각각 3개와 5개의 군집이 획득되었, 이 군집의 중심은 각각 0.17, 0.33, 0.50과 0.22, 0.25, 0.31, 0.38, 0.57이었다. 이러한 결과를 바탕으로 표 4와 같이 각 방에 대한 Q-table의 상태가 생성되었다. Java를 통해 Q-table을 포함한 Q-learning 에이전트가 구현되어 BLE Tag Receiver에 설치되었다.

표 4

State	Small Room		Large Room	
	Near Boundary (m)	Far Boundary (m)	Near Boundary (m)	Far Boundary (m)
1	1.00	1.50	1.93	2.50
2	1.17	1.67	2.12	2.72
3	1.33	1.83	2.19	2.75
4	1.50	2.00	2.25	2.81
5	-	-	2.28	2.88
6	-	-	2.50	3.07

[0114]

[0115] 거리 추정 모델의 성능을 평가하기 위해 모델(즉, iBeacon, 선형 회귀 모델, 딥 러닝 모델)에 따라 달라지는 추정 거리를 비교했다. 도 5a, 5b는 작은 방과 큰 방 각각에 대한 거리 추정 모델을 보여준다. iBeacon 모델은 방의 유형에 관계없이 고정되어 있지만 선형 회귀 및 딥 러닝 모델은 훈련 데이터 세트에 따라 변경된다. 따라서 각 도면에는 동일한 iBeacon 모델이 표시되지만 서로 다른 선형 회귀 및 딥 러닝 모델이 표시된다. 도 5a 및 도 5b에서 추정 거리는 추정 모델에 관계없이 RSSI가 증가함에 따라 기하급수적으로 감소하는 경향이 있다. 그러나

추정 거리의 차이는 RSSI가 감소할수록 커진다. 이는 모델마다 RSSI와 추정 거리 간의 관계를 다르게 정의하기 때문이다. 특히 iBeacon 모델의 추정 거리는 선형 회귀 및 딥 러닝 모델보다 RSSI의 변화에 더 큰 영향을 받는다. 선형 회귀 모델과 딥 러닝 모델을 비교하면 후자가 RSSI의 영향을 덜 받는다.

[0116] 표 5와 6은 각각 작은 방과 큰 방에 대한 추정 거리와 RMSE(Root Mean Square Error)를 나타낸다. 추정 거리를 얻기 위해 테스트 데이터 세트가 두 모델에 입력되었다. 거리 추정에 사용된 RSSI 데이터셋은 서로 다른 크기의 방에서 수집되었기 때문에 동일한 실제 거리에 대한 추정 거리는 각 테이블에서 다르게 표시되었다. 선형 회귀 모델은 환경에 따라 데이터 값이 달라지는 LD를 기반으로 구축되었기 때문에 iBeacon 모델보다 더 정확한 추정 결과를 제공하였다. 따라서 표에서 선형 회귀 모델은 평균적으로 iBeacon 모델보다 추정 오차(즉, 실제 거리와 추정 거리의 차이)가 작다. 결과적으로 선형 회귀 모델의 RMSE는 iBeacon 모델의 RMSE보다 작다. 정량적으로 작은 방과 큰 방에 대해 선형 회귀 모델은 iBeacon 모델에 비해 각각 76.18% 및 65.17% 더 작은 RMSE를 얻었다. 선형 회귀 모델과 유사하게 방의 유형에 따라 다른 딥 러닝 모델이 구축된다. 그러나 딥 러닝 모델은 여러 RSSI 입력을 사용하여 거리를 추정한다. 따라서 선형 회귀 모델에 비해 RSSI 변동성의 영향을 덜 받았다. 특히, 각 방에 대해 딥 러닝 모델은 선형 회귀 모델보다 각각 1.23% 및 36.40% 더 작은 RMSE를 갖는다. RMSE는 이하 수학적 식 23을 통해 산출될 수 있다.

수학적 식 23

$$RMSE = \sqrt{\frac{\sum_{i=1}^N \|y(i) - \hat{y}(i)\|^2}{N}}$$

[0118] 여기서, N은 테스트 데이터 세트의 샘플의 수이고, y(i)는 i번째 데이터 샘플에 대한 실제 거리를 나타내며, $\hat{y}(i)$ 는 i번째 데이터 샘플에 대한 추정 거리를 나타낸다.

표 5

Actual Distance (m)	Estimated Distance (m)			RMSE		
	iBeacon	Linear Regression	Deep Learning	iBeacon	Linear Regression	Deep Learning
0.5	0.76	0.86	0.48	0.30	0.37	1.14
1	1.64	1.21	1.03	0.76	0.26	0.92
1.5	2.52	1.36	1.66	1.23	0.18	0.78
2	2.55	1.49	2.36	7.44	1.00	0.70
2.5	7.91	2.59	2.52	2.69	0.53	0.66
3	4.75	2.03	2.59	9.95	1.15	0.66
4	10.22	2.93	2.69	1.43	2.19	0.71
Average	-	-	-	3.40	0.81	0.80

표 6

Actual Distance (m)	Estimated Distance (m)			RMSE		
	iBeacon	Linear Regression	Deep Learning	iBeacon	Linear Regression	Deep Learning
0.5	0.68	0.71	0.44	0.23	0.23	1.18
1	1.59	1.23	1.09	0.74	0.33	0.95
1.5	2.40	1.61	1.54	0.46	0.13	0.79
2	1.91	1.40	2.14	2.84	0.86	0.69
2.5	4.16	2.37	3.51	5.03	1.37	0.63
3	5.68	2.96	2.57	5.27	1.42	0.61
4	6.98	3.48	3.62	1.77	1.04	0.64
5	5.68	3.02	4.29	2.25	2.08	0.71
6	5.82	3.05	6.15	14.39	2.73	0.85
7	18.27	7.66	6.34	4.95	3.01	0.96
Average	-	-	-	3.79	1.32	0.80

- [0121] 본 개시의 시스템의 QPZA의 타당성을 검증하기 위해 크기가 1m로 고정된 고정 보류 구간의 정확도와 QPZA의 정확도가 비교되었다. 실험에서 정확도는 테스트 데이터 세트에 포함된 모든 데이터 샘플에서 올바른 추정치(즉, True 및 False 추정치)의 비율을 나타낸다. 도 6의 (a) 내지 (f)는 서로 다른 거리 추정 모델을 사용하여 작은 방과 큰 방에 대한 근접 분류의 정확도를 보여준다. 수치는 실제 거리가 보류 구간(Pending Zone)의 경계에 가까울 때 정확도가 감소함을 나타낸다. RSSI 변동성에 의한 거리 추정 모델의 오차는 실제 거리가 Pending Zone의 경계에 가까울수록 근접 분류에 더 큰 영향을 미치기 때문이다.
- [0122] 큰 방의 경우 방의 크기가 커질 때 다중경로 페이딩 등의 외부요인에 의한 RSSI의 영향이 크기 때문에 작은 방에 비해 평균 추정 거리의 오차가 컸다. 따라서 큰 방에 비해 작은 방에서 더 높은 정확도를 얻었다. 평균적으로 작은 방에 대한 고정 보류 구간 및 QPZA의 정확도는 각각 90.51% 및 96.55%였다. 반면 이들의 정확도는 큰 방의 경우 각각 89.83%와 94.4%였다.
- [0123] 도면에서 보는 바와 같이 거리 추정 모델의 RMSE가 낮을수록 평균 정확도가 높다. 특히 작은 방에서 iBeacon, 선형 회귀 및 딥 러닝 모델은 각각 88.87%, 90.50% 및 91.14%의 정확도를 나타낸다. 큰 방에서는 각각 평균 93.93%, 95.87%, 96.69%의 정확도를 달성했다. 이는 정확한 거리 추정이 근접 분류의 정확도를 높일 가능성이 높기 때문이다.
- [0124] 전반적으로, QPZA는 노이즈 레벨에 따라 Pending Zone의 크기를 적응적으로 조정하기 때문에 고정된 Pending Zone에 비해 더 나은 성능을 보였다. 평균적으로 QPZA는 각 방에 대해 고정 보류 구간보다 6.68% 및 5.13% 더 높은 정확도를 가진다. 각각의 방에 대하여, 실제 거리가 기준 거리와 동일한 경우(즉, 작은 방의 경우 1.5m, 큰 방의 경우 2.5m), QPZA는 각 방의 고정 보류 구간에 비해 13.67% 및 9.70% 더 높은 정확도를 달성했다.
- [0125] QPZA의 보상 함수가 근접성 분류의 정확도에 미치는 영향을 조사하기 위해 노이즈 수준 계수(즉, 방정식 (17) 및 (18)의 k)를 각 방에 대해 1, 2 및 4 사이에서 변경했다. 그림 7a, 7b는 노이즈 레벨 계수에 따른 근접 분류의 정확도를 보여준다. 방 유형에 관계없이 $k=2$ 일 때 근접성 분류의 평균 정확도가 가장 높았다. 추정 거리와 기준 거리의 차이가 노이즈 레벨의 절반 수준에 가까웠기 때문이다. $k=1$ 일 때 근거리 및 원거리 경계에 대한 각 보상은 $k=2$ 에 비해 각각 -1 및 1 쪽으로 더 편향되었다. 반대로 $k=4$ 일 때 각각 1과 -1 쪽으로 더 편향되었다. $k=2$ 및 $k=4$ 인 경우 대기 중인 영역의 크기가 적절하지 않아 정확도가 떨어지는 근접성 분류가 발생한다. 정량적으로 $k=2$ 에 대한 근접 분류의 정확도는 $k=1$ 및 $k=4$ 에 비해 각각 9.10% 및 7.34% 더 높았다.
- [0126] 한편, 이상에서 설명된 다양한 실시 예들은 서로 저촉되지 않는 한 둘 이상의 실시 예가 함께 구현될 수 있다.
- [0127] 한편, 이상에서 설명된 다양한 실시 예들은 소프트웨어(software), 하드웨어(hardware) 또는 이들의 조합된 것을 이용하여 컴퓨터 또는 이와 유사한 장치로 읽을 수 있는 기록 매체 내에서 구현될 수 있다.
- [0128] 하드웨어적인 구현에 의하면, 본 개시에서 설명되는 실시 예들은 ASICs(Application Specific Integrated Circuits), DSPs(digital signal processors), DSPDs(digital signal processing devices), PLDs(Programmable logic devices), FPGAs(field programmable gate arrays), 프로세서(processor), 제어기(controller), 마이크로 컨트롤러(micro-controllers), 마이크로 프로세서(microprocessor), 기타 기능 수행을 위한 전기적인 유닛(unit) 중 적어도 하나를 이용하여 구현될 수 있다.
- [0129] 일부의 경우에 본 명세서에서 설명되는 실시 예들이 프로세서 자체로 구현될 수 있다. 소프트웨어적인 구현에 의하면 본 명세서에서 설명되는 절차 및 기능과 같은 실시 예들은 별도의 소프트웨어 모듈들로 구현될 수 있다. 상술한 소프트웨어 모듈들 각각은 본 명세서에서 설명되는 하나 이상의 기능 및 작동을 수행할 수 있다.
- [0130] 한편, 상술한 본 개시의 다양한 실시 예들에 따른 전자 장치의 처리동작을 수행하기 위한 컴퓨터 명령어(computer instructions)는 비일시적 컴퓨터 판독 가능 매체(non-transitory computer-readable medium)에 저장될 수 있다. 이러한 비일시적 컴퓨터 판독 가능 매체에 저장된 컴퓨터 명령어는 특정 기기의 프로세서에 의해 실행되었을 때 상술한 다양한 실시 예에 따른 전자 장치(ex. Tag Receiver, 서버 등)의 처리 동작을 상술한 기기가 수행하도록 한다.
- [0131] 비일시적 판독 가능 매체란 레지스터, 캐쉬, 메모리 등과 같이 짧은 순간 동안 데이터를 저장하는 매체가 아니라 반영구적으로 데이터를 저장하며, 기기에 의해 판독(reading)이 가능한 매체를 의미한다. 구체적으로는, 상술한 다양한 어플리케이션 또는 프로그램들은 CD, DVD, 하드 디스크, 블루레이 디스크, USB, 메모리카드, ROM 등과 같은 비일시적 판독 가능 매체에 저장되어 제공될 수 있다.
- [0132] 또한, 이상에서는 본 발명의 바람직한 실시 예에 대하여 도시하고 설명하였지만, 본 발명은 상술한 특성의 실시

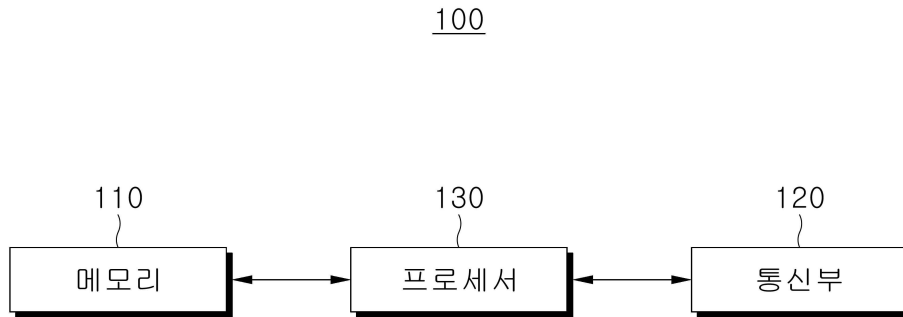
예에 한정되지 아니하며, 청구범위에서 청구하는 본 발명의 요지를 벗어남이 없이 당해 발명이 속하는 기술분야에서 통상의 지식을 가진 자에 의해 다양한 변형실시가 가능한 것은 물론이고, 이러한 변형실시들은 본 발명의 기술적 사상이나 전망으로부터 개별적으로 이해해서는 안 될 것이다.

부호의 설명

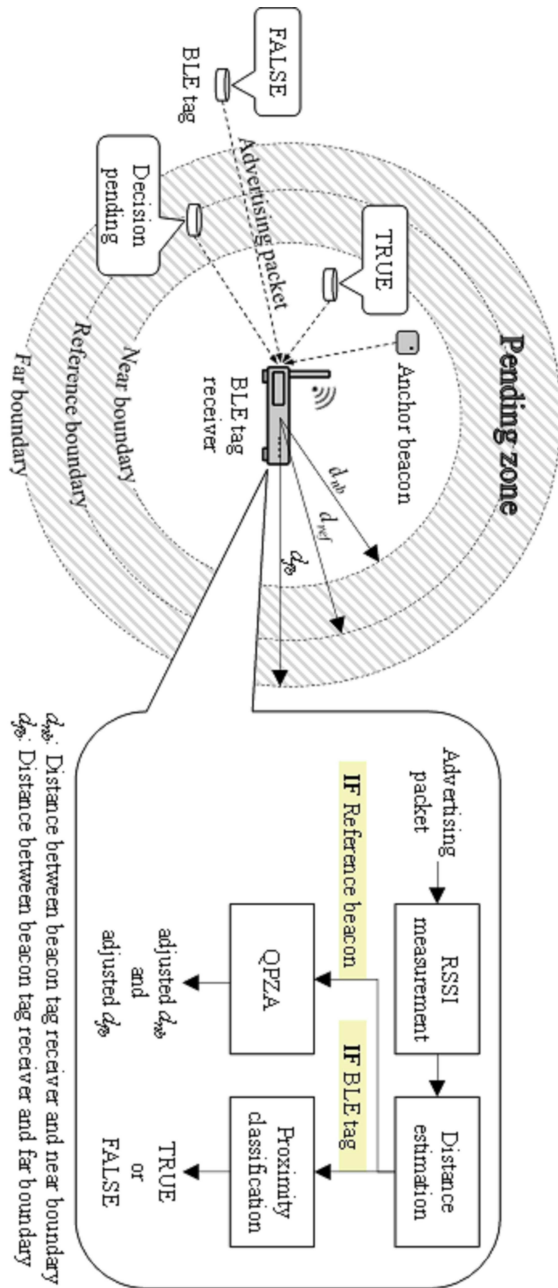
100: Tag Receiver

도면

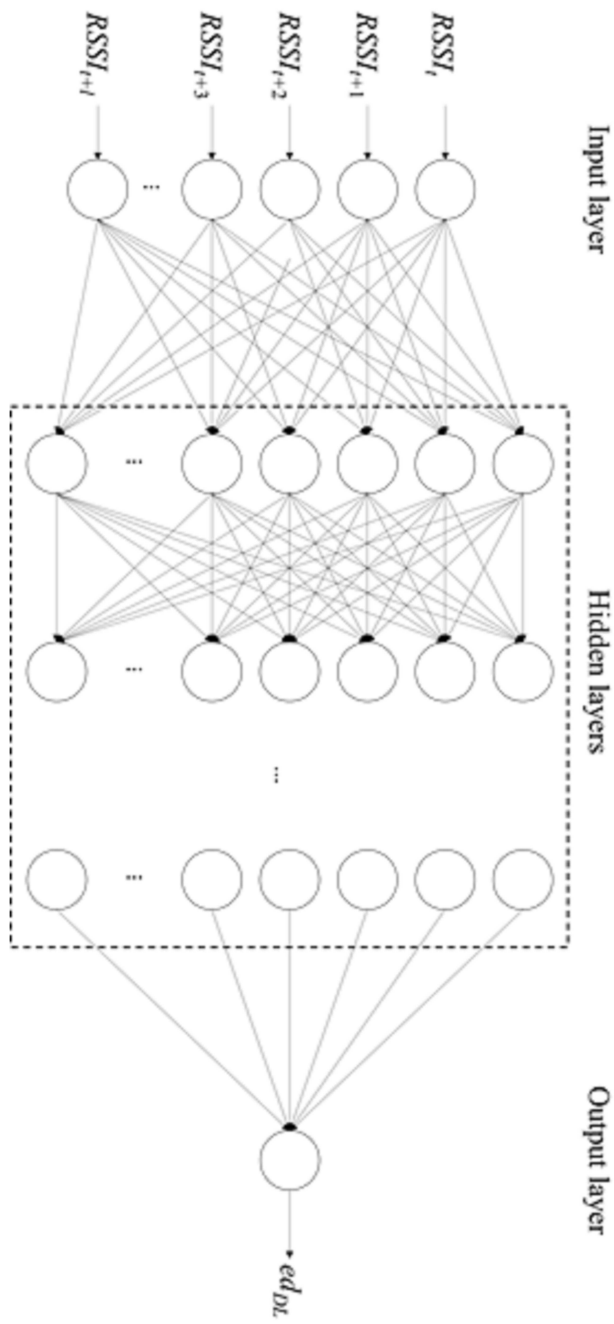
도면 1a



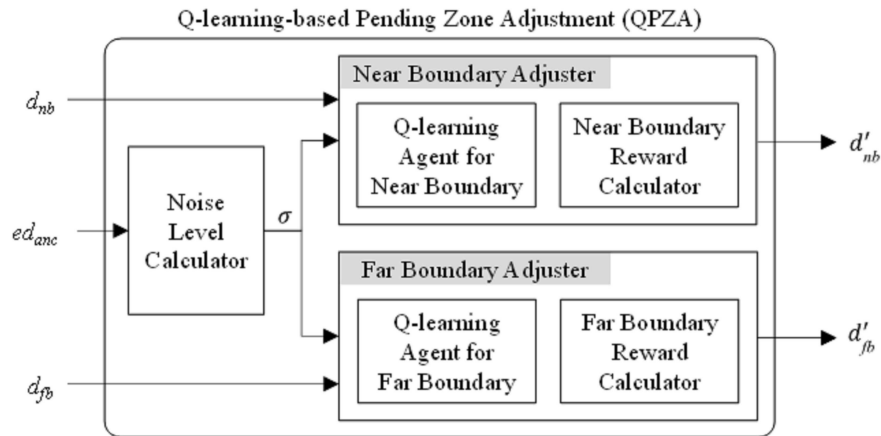
도면1b



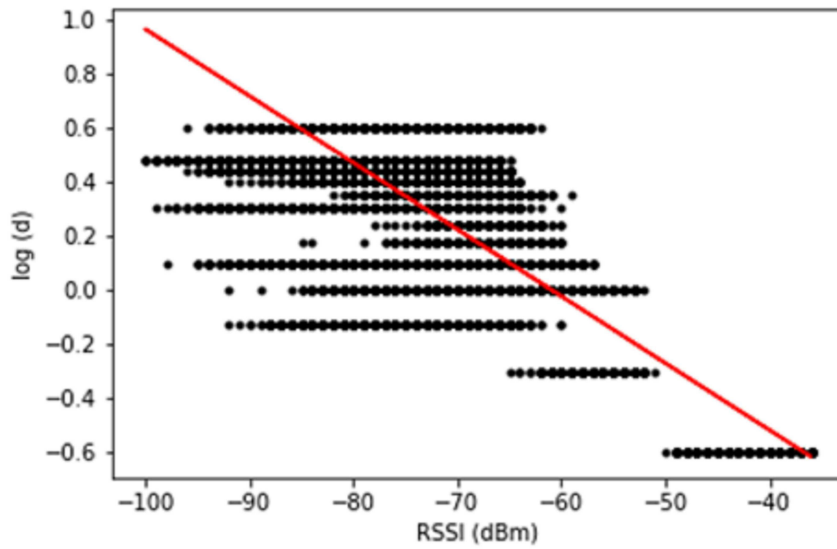
도면2



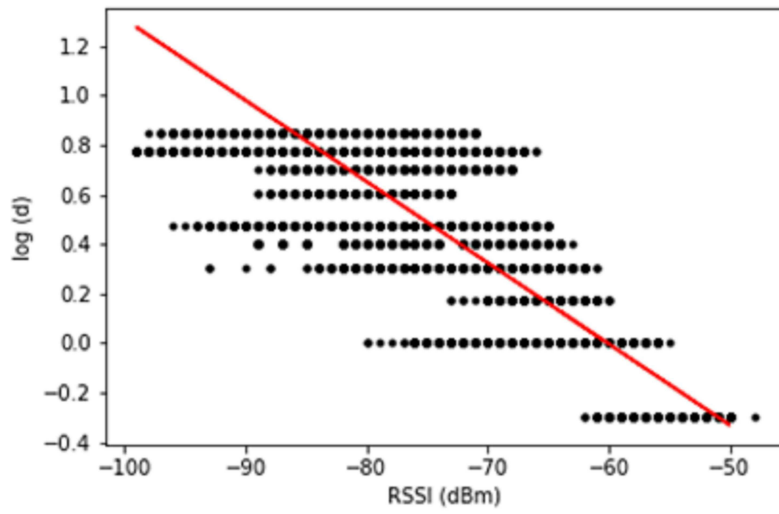
도면3



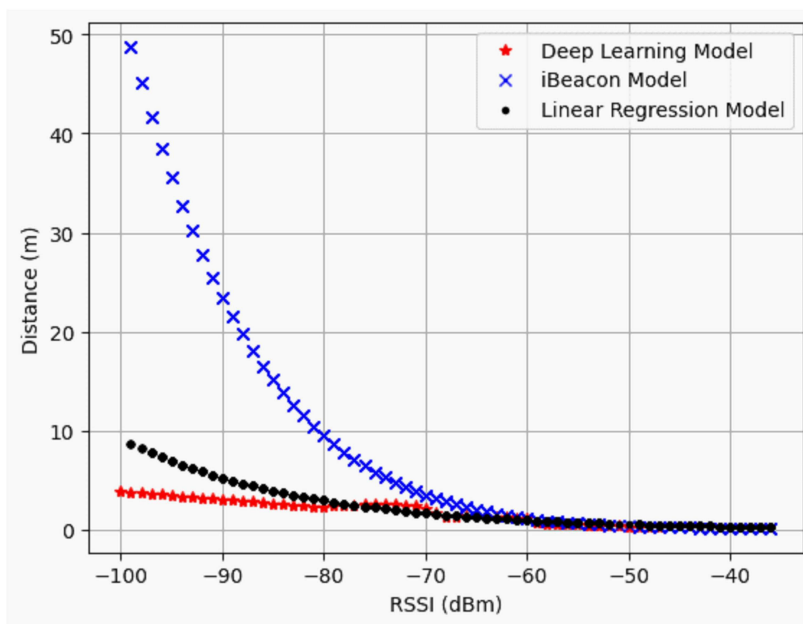
도면4a



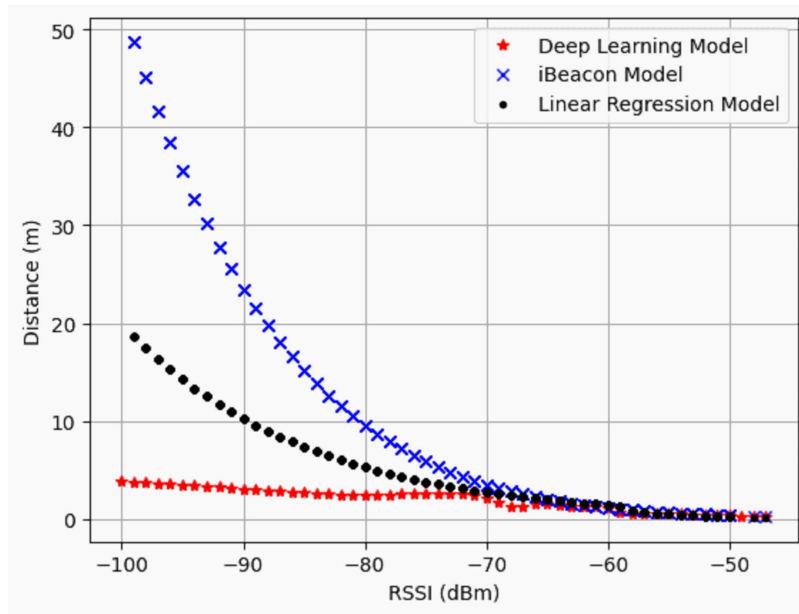
도면4b



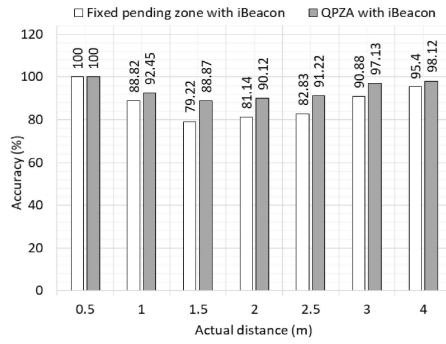
도면5a



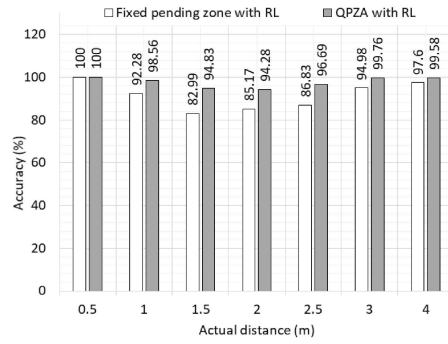
도면5b



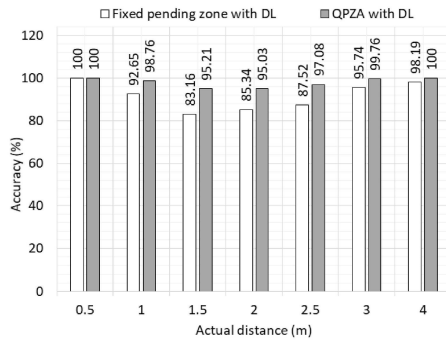
도면6



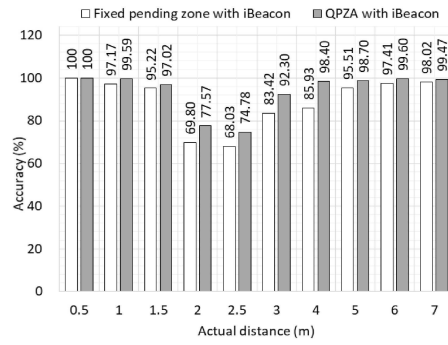
(a)



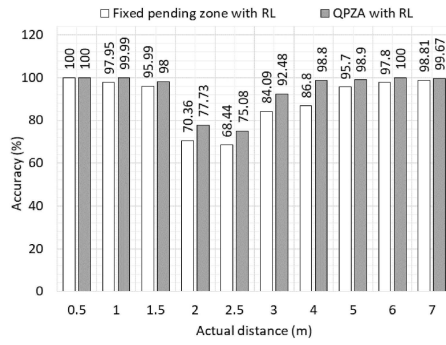
(b)



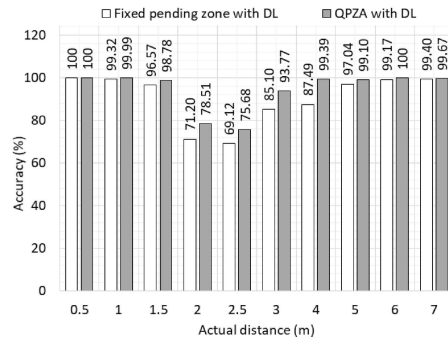
(c)



(d)



(e)



(f)