

온보드 고속 목표물 추적을 위한 FPGA 기반 양자화 병렬 입자 필터링

김나연*¹, 이헌철*², 권혁훈**¹, 최원석**², 정보라**³

Quantized Parallel Particle Filtering on FPGA for Onboard High-Speed Target Tracking

Nayeon Kim*¹, Heoncheol Lee*², Hyuckhoon Kwon**¹, Wonseok Choi**², and Bora Jung**³

This work was supported in part by Theatre Defense Research Center funded by Defense Acquisition Program Administration under Grant UD240002SD, and in part by the IITP(Institute of Information & Communications Technology Planning & Evaluation)-ITRC(Information Technology Research Center) grant funded by the Korea government(Ministry of Science and ICT)(IITP-2025-RS-2024-00438430, 50%), and in part by the Gyeongsangbuk-do RISE (Regional Innovation System & Education) project (scale-up unit)

요약

본 연구는 온보드 환경에서 고속 탄도 목표물과 같은 비선형/비가우시안 표적을 실시간으로 추적하기 위한 파티클 필터 가속화 방안을 제안한다. 기존의 순차적 방식의 파티클 필터는 대량의 파티클 연산으로 인해 실시간 처리에 한계가 있다. 이러한 문제를 해결하기 위해 본 논문에서는 양자화 기법과 병렬 처리를 결합한 새로운 하드웨어 가속 구조를 제안한다. Zynq FPGA 플랫폼을 기반으로 연산 병목 연산 구간인 Model Propagation과 리샘플링을 위한 IP를 설계하였으며, 배정밀도 연산을 INT16 정수 연산으로 변환하는 클리핑 기반의 대칭 양자화 기법을 적용하고 파이프라이닝 및 언롤링 기법을 적용하여 연산 효율을 극대화했다. 시뮬레이션 결과, 제안된 구조는 전체 실행 시간을 48% 단축하는 데 성공했으며, 양자화로 인한 오차는 최종 추적 성능에 미미한 영향을 미쳐 기존과 동일한 수준의 정확도를 유지했다.

Abstract

This paper proposes a hardware-accelerated particle filter architecture for real-time, high-speed target tracking in on-board system, combining clipping-based symmetric quantization with parallel processing. To overcome the bottlenecks of conventional sequential methods, we designed custom IP cores on a Zynq FPGA for Model Propagation and Resampling, optimized with pipelining and unrolling techniques. The proposed architecture achieved a 48% reduction in total execution time compared to a Processing System (PS) implementation. This work validates our design as an effective solution that balances real-time performance and precision, showing strong potential for next-generation AI processors.

Keywords

Particle Filter, FPGA, quantization, parallelization

* 국립금오공과대학교 IT융복합공학과(*² 교신저자) · Received: Nov. 10, 2025, Revised: Dec. 12, 2025, Accepted: Dec. 15, 2025
- ORCID¹: <https://orcid.org/0009-0003-5000-582X> · Corresponding Author: Heoncheol Lee
- ORCID²: <https://orcid.org/0000-0003-2962-3474> Dept. of IT Convergence Engineering, School of Electronic Engineering
** LIG넥스원 Kumoh National Institute of Technology, Korea
- ORCID¹: <https://orcid.org/0000-0003-2773-0145> Tel.: +82-54-478-7476, Email: hclee@kumoh.ac.kr
- ORCID²: <https://orcid.org/0009-0000-2663-4421>
- ORCID³: <https://orcid.org/0009-0009-7777-7954>

1. 서 론

방어 체계에서 탄도 궤적을 갖는 고속 목표물을 정확하게 실시간 예측하는 것은 요격 성공을 결정짓는 가장 중요한 요소이다. 고속 탄도 목표물의 상태를 추정하기 위해선 다양한 센서를 활용한 측정 데이터를 활용해야 한다. 하지만 실제 전장 환경의 측정 데이터는 비선형 및 비가우시안 잡음이 존재하기 때문에 그대로 사용하기엔 어려움이 있다. 따라서 확률적 필터링 알고리즘 기법을 통해 목표물의 상태를 안정적으로 추정하는 연구들이 다양하게 진행되었다[1]-[3].

대표적인 확률적 필터링 알고리즘은 칼만 필터이다. 칼만 필터는 단순하며 선형 시스템에서 최적의 성능을 보장하며 실시간 시스템에 적합하지만 비선형 시스템에는 부적합하다[4][5]. 이러한 한계를 극복하고자 확장 칼만 필터, 무향 칼만 필터, 파티클 필터와 같은 다양한 필터들이 탄생했다. 먼저 확장 칼만 필터는 미분 과정을 통해 비선형 함수 자체를 선형화한다. 이는 계산 비용이 적다는 장점이 있지만 이상치에 대한 내성이 떨어진다는 단점이 존재한다[6][7]. 무향 칼만 필터는 시그마 포인트를 선택하고 이 포인트들을 비선형 함수에 적용한 후, 그 결과로부터 가우시안 분포를 재계산함으로써 비선형 시스템을 처리한다[8]. 그러나 이는 생성된 모든 시그마 포인트를 복잡한 비선형 함수에 적용해야 하며 시그마 포인트 수가 증가할수록 정확도는 높아지지만 계산 시간도 증가하여 실시간 추정에 적합하지 않다. 마지막으로 파티클 필터는 비선형과 비가우시안 시스템에 효과적인 필터링이다. 입자 필터는 샘플링 기반 방법을 사용하여 비선형 시스템 내에서 입자가 자유롭게 이동할 수 있도록 한다. 이는 복잡한 동적 모델을 처리하고 다양한 상태 확률 분포를 표현하는 데 효과적이다[9].

따라서 복잡한 노이즈와 고차원 상태 공간을 가진 탄도 표적 추적에서 입자 필터를 활용하면 불확실성을 줄이고 안정적인 추정 결과를 보장할 수 있다. 그러나 입자 필터의 주요 문제점은 높은 계산 비용으로 인해 고속 탄도 목표물을 실시간으로 추적하기 어렵다는 점이다. 고속 표적을 실시간으로

정확하게 추적하려면 많은 수의 입자가 필요하며 입자 수를 늘리면 성능은 향상되지만 계산 복잡성도 증가한다. 따라서 추적 정확도와 빠른 실행 시간을 동시에 달성하기 위해 다양한 하드웨어 기반 가속 기법을 적용해야 한다.

GPU(Graphics Processing Unit)는 알고리즘 가속을 위해 가장 널리 사용되는 하드웨어 가속기 중 하나이며 GPU 기반 입자 필터 가속에 관한 많은 연구가 수행되었다[10]-[12]. [10]은 실행 시간을 크게 단축했으며 실행 시간 측면에서 상당한 성과를 거두었다. 그러나 GPU는 많은 전력을 소모하며 계산 부하가 증가함에 따라 발열량도 함께 증가한다. 결과적으로 제한된 전력 자원으로 작업을 수행하는 방위 산업에서 GPU 사용이 어렵다. 또한 알고리즘의 가속화를 위해 FPGA(Field-Programmable Gate Array) 기반 알고리즘 가속화 연구도 진행되었다[13]-[16]. FPGA는 플랫폼에 맞게 맞춤화할 수 있어 불필요한 자원 사용을 최소화하고 지정된 작업에 최적화된 논리로 작업을 실행할 수 있다.

최근 AMD사의 Versal과 같은 저전력 AI 프로세서 활용함으로써 벡터 기반의 연산 처리가 가능하고 데이터 전송 시간을 최적화하여 고속 데이터 전송이 가능하다는 특징을 갖는다. 하지만 이러한 프로세서들은 대부분 정수형 기반의 연산을 지원하기 때문에 기존에 정확도를 위해 사용되던 실수형 기반의 데이터를 변환할 필요가 있다[17]. 또한 전력 및 자원이 제한된 온보드 환경에서는 계산 오버헤드를 줄이고 성능을 최적화하기 위해 양자화 기법(Quantization)이 사용된다. 특히 대규모 연산을 수반하는 딥 러닝 분야에서는 계산 효율성을 위한 양자화 연구가 활발히 진행됐다. 입자 필터 내 대규모 행렬 곱셈에 사용되는 데이터에 수식이 간결하고 이상치에 대한 오버플로우를 방지하여 오차가 누적되는 것을 최소화하는 클리핑 기반의 대칭 양자화를 적용함으로써 성능을 유지하면서 더 빠른 실행 시간과 효율적인 자원 사용을 달성할 수 있다.

파티클 필터의 양자화를 고려한 연구[18]는 센서 성능 저하와 측정치 양자화가 동시에 발생하는 환경에서 다시 우도 함수를 계산하여 정확도를 향상했다. 해당 연구는 저성능 센서에도 정확도가 향상

된다는 장점이 있지만 성능 저하와 양자화 오류를 모두 고려한 복잡한 우도 함수를 계산함으로써 기존보다 연산량이 더 증가한다는 한계가 있다. [19]는 파티클 필터의 리샘플링 단계를 고정소수점으로 하드웨어 구현할 때 양자화 오류로 인해 리샘플링 후 입자 수가 의도한 것과 달라지는 문제를 해결하기 위한 보상 알고리즘을 제안하며 고정소수점 연산 시 발생하는 부작용을 해결하고자 했다. 이는 리샘플링 단계의 구현 문제에 국한되어있다. 이 외에도 대부분 양자화된 데이터를 처리하기 위해 파티클 필터를 사용하는 연구가 진행되었다[20]-[22].

본 연구에서는 파티클 필터의 가속화를 위해 FPGA에서 효율적으로 수행할 수 있는 양자화 병렬 입자 필터링을 제안한다. 제안된 방법은 향후 적용될 저전력 AI 프로세서와 같은 온보드 장치의 FPGA와의 호환성을 높이기 위해 기존의 64비트의 실수형 데이터 대신 정수형의 16비트 데이터를 사용하기 위한 클리핑 기반의 대칭 양자화 기법을 적용한다. 또한 변환된 정수형 16비트 데이터를 기반으로 파티클 필터의 연산을 처리하며 최종적으로 다시 역양자화 기법(Dequantization)을 적용하여 기존 실수형 데이터의 결과와 큰 오차 없이 안정적으로 고차원의 고속 탄도 목표물을 추적함을 보여준다.

2장에서는 파티클 필터를 활용한 고속 탄도 목표물의 상태 추적과 실시간성의 한계를 정의하고, 3장에서 제안하는 파티클 필터의 양자화 기법과 이를 적용한 알고리즘의 FPGA 기반 병렬화 기법을 설명한다. 마지막으로 4장에서는 파티클 필터를 적용한 고속 탄도 목표물의 상태 추정 시뮬레이션 결과를 제시하고, 수행시간 분석 및 자원 사용량 평가를 진행하며 결론을 제시한다.

II. 문제점 기술

탄도 표적 추적에서 가장 중요한 과제는 표적의 현재 상태를 추정하는 것이다. 탄도 표적 추적 알고리즘은 표적의 궤적을 얼마나 정확하게 예측할 수 있는지에 따라 평가된다. 이를 위해 재진입 단계 동안 표적의 예상 궤적을 대기 저항을 고려하여 계산해야 한다. 탄도 표적이 대기로 재진입할 때 공기역학적 힘을 고려해야 한다.

본 논문에서는 탄도 표적을 3차원 직교 좌표계상의 점 질량으로 그림 1과 같이 모델링한다. 추력은 표적만 고려되며 탄도 표적의 질량이 일정하므로 0으로 가정한다. 대기권 재진입 시 탄도 표적의 고속 특성으로 인한 양력은 항력에 비해 무시할 수 있다. 따라서 재진입 단계의 궤적은 다음과 같은 비선형 동역학 방정식으로 표현된다.

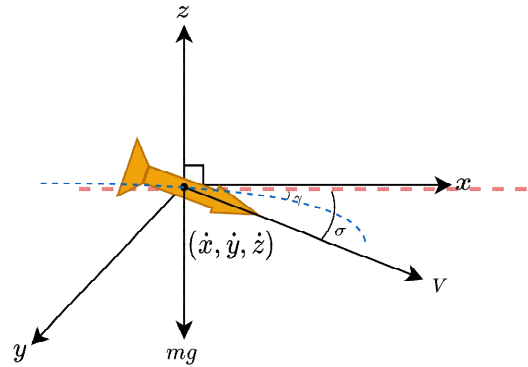


그림 1. 직교 좌표계에서 탄도 물체 궤적
Fig. 1. Trajectory of the ballistic object in the cartesian coordinate system

본 논문에서는 탄도 표적을 3차원 직교 좌표계상의 점 질량으로 그림 1과 같이 모델링한다. 추력은 표적만 고려되며 탄도 표적의 질량이 일정하므로 0으로 가정한다. 대기권 재진입 시 탄도 표적의 고속 특성으로 인한 양력은 항력에 비해 무시할 수 있다. 따라서 재진입 단계의 궤적은 다음과 같은 비선형 동역학 방정식으로 표현된다.

$$\dot{x} = V \cos \gamma \cos \chi \quad (1)$$

$$\dot{y} = V \cos \gamma \sin \chi$$

$$\dot{z} = -V \sin \gamma$$

$$\dot{V} = \frac{(T - F_D - mg \sin \gamma)}{m} \quad (2)$$

$$\dot{\gamma} = \frac{(F_L \cos \theta - mg \cos \gamma)}{m}$$

$$\dot{\chi} = \frac{F_L \sin \theta}{m V \cos \gamma}$$

$$F_d = \frac{1}{2} \rho V^2 \cdot C_d \cdot S \quad (3)$$

$$F_L = \frac{1}{2} \rho V^2 \cdot C_L \cdot S$$

x, y, z 는 탄도 표적의 위치, V 는 속도 그리고 T 는 추력이다. 공력 F_d, F_L 은 각각 항력과 양력이며, 이때 ρ 는 공기 밀도 상수, C_d, C_L 은 항력 및 양력 계수, S 는 기준 면적이다. 또한 γ, χ 는 각각 비행 경로각, 기수 방향각을 나타내며 m, g 는 표적의 질량과 중력 상수이다. 마지막으로 θ 는 벡터각으로 이는 양력의 방향을 결정한다.

2.1 Singer model 및 측정 모델

동적 표적의 추적은 다양한 모델을 사용하여 해결할 수 있으며 그중 싱어 모델이 많이 사용되는 모델 중 하나이다. 싱어 모델은 단순한 궤적을 따르는 표적과는 달리 빠르고 급격하게 변화하며 움직이는 동적 표적을 추적하는 데 특히 적합하다. 결과적으로 이 모델은 표적의 상태를 정밀하게 추정할 수 있게 해주며 시간에 따른 움직임의 갑작스러운 변화를 효율적으로 추적한다. 표적의 가속도는 평균이 0이라고 가정되며, 1차 정상 마르코프 과정으로 표현되며 이는 다음과 같다.

$$\dot{x} = Hx + Gw \tag{4}$$

$$x = [p_x \ p_y \ p_z \ v_x \ v_y \ v_z \ a_x \ a_y \ a_z]^T \tag{5}$$

$$H = \begin{bmatrix} 0_3 & I_3 & 0_3 \\ 0_3 & 0_3 & I_3 \\ 0_3 & 0_3 & -I_3/\tau \end{bmatrix}, G = \begin{bmatrix} 0_3 \\ 0_3 \\ I_3 \end{bmatrix} \tag{6}$$

x 는 목표물의 상태 벡터를 나타내며, 레이더 시스템에 의해 측정된 탄도 표적의 위치, 속도, 가속도이다. 행렬 H, G 는 3x3 단위 행렬을 포함하며, τ 는 가속도의 시상수다. H 는 시스템 행렬로, 다음 상태에 현재 상태가 어떻게 영향을 끼칠지 결정하고, G 와 w 는 불확실한 변화를 나타내는 노이즈와 백색 잡음을 나타낸다.

$$\begin{aligned} x_k &= \Phi_{k-1}x_{k-1} + w_{k-1}, \\ w_{k-1} &\sim N(0, Q_k), \\ \Phi_k &\simeq I + H\Delta t \end{aligned} \tag{7}$$

$$Q_k \simeq S_w Q_0 \simeq S_w \begin{bmatrix} \frac{\Delta t^5}{20} I_3 & \frac{\Delta t^4}{8} I_3 & \frac{\Delta t^3}{6} I_3 \\ \frac{\Delta t^4}{8} I_3 & \frac{\Delta t^3}{3} I_3 & \frac{\Delta t^2}{2} I_3 \\ \frac{\Delta t^3}{6} I_3 & \frac{\Delta t^2}{2} I_3 & \Delta t I_3 \end{bmatrix} \tag{8}$$

식 (7)-(8)은 시간 방정식을 나타낸다. Φ_k 는 상태 전이 행렬, w 는 평균이 0인 백색 가우시안 잡음, Δt 는 이산 시간 시스템에서 연속적인 상태 사이의 시간 간격을 나타낸다. Q_k 는 공분산, S_w 는 파워 스펙트럼 밀도, Q_0 는 백색 잡음 저크 모델이다. 저크 모델의 적분을 사용한다는 것은 시간이 지남에 따라 가속도가 변화함을 나타낸다. 또한 추적되는 표적과 레이더 사이의 상대적인 위치를 나타내기 위해 아래의 수식을 사용한다.

$$[x_m \ y_m \ z_m]^T = [x \ y \ z]^T - [x_r \ y_r \ z_r]^T \tag{9}$$

$$\begin{bmatrix} q_\theta \\ q_\psi \\ q_R \end{bmatrix} = \begin{bmatrix} \tan^{-1}\left(\frac{z_m}{\sqrt{x_m^2 + y_m^2}}\right) + n_{m,\theta} + n_\theta \\ \tan^{-1}\left(\frac{y_m}{x_m}\right) + n_{m,\psi} + n_\psi \\ \sqrt{x_m^2 + y_m^2 + z_m^2} + n_R \end{bmatrix} \tag{10}$$

x_m, y_m, z_m 은 표적과 레이더의 상대 위치, x, y, z 는 데카르트 좌표계에서의 표적 위치, x_r, y_r, z_r 은 레이더의 좌표다. 상대 정보를 사용하여 각도와 거리 q_θ, q_ψ, q_R 는 식 (10)에서 정의된다. 실제 특성을 고려해 레이더 수신기 잡음 n_θ, n_ψ, n_R 과 측정 잡음 $n_{m,\theta}, n_{m,\psi}$ 을 포함한다. 측정 잡음으로는 비가우시안 특성의 글린트 잡음(Glint noise)이 생성된다.

2.2 Particle Filter의 실시간성 문제

고속 탄도 목표물을 추적하기 위해 파티클 필터를 적용하려면 계산 속도와 정확한 추정 성능이 모두 중요하다. 그러나 파티클 필터의 주된 단점은 각 입자의 위치가 갱신되어야 하고 그 가중치가 확률 밀도 함수에 기반하여 계산되어야 하므로 계산 비용이 입자의 수에 비례하여 증가한다는 것이다. 입자의 수를 늘리면 더 안정적인 추정 성능을 얻을

수 있지만 계산 속도가 느려져 성능과 계산 비용 사이에 상충 관계가 발생하기 때문에 파티클 필터를 사용하면서 좋은 성능을 내는 동시에 실시간 추적을 가능하게 하는 것은 어렵다[10][13].

이러한 문제를 극복하기 위해서는 파티클 필터의 가속이 필수적이다. FPGA를 활용하면 순차적인 계산의 병렬화가 가능해져 전체 실행 시간을 크게 줄일 수 있다. 그러나 모든 연산이 FPGA 가속으로부터 동일한 이점을 얻는 것은 아니므로 어떤 계산이 병렬화에 적합한지 식별하고 각 프로세스에 적절한 방법을 적용하는 것이 중요하다.

본 연구에서 다루는 파티클 필터의 수행시간을 분석한 프로파일링 결과는 그림 2와 같으며 리샘플링, 모델 전파, 가중치 연산 순으로 수행시간이 소요된다. 해당 수행시간은 입자 개수 2000개를 기준으로 측정되었으며, 파티클 필터의 높은 연산량의 이유는 각 입자마다 개별적으로 처리하는 반복 계산을 수행하기 때문에 연산량이 입자 수와 비례하여 증가한다. 또한 양자화 과정이 포함되면서 데이터 변환을 위한 수행시간이 추가된다. 모델 전파와 가중치 연산의 경우, 각 입자마다 행렬 곱이 진행되기 때문에 높은 계산 복잡도를 갖는다. 리샘플링의 경우는 기존의 가중치와 랜덤 변수를 포함하며 파티클 개수의 2배를 정렬하기 때문에 입자 수에 비례하여 수행시간이 증가한다.

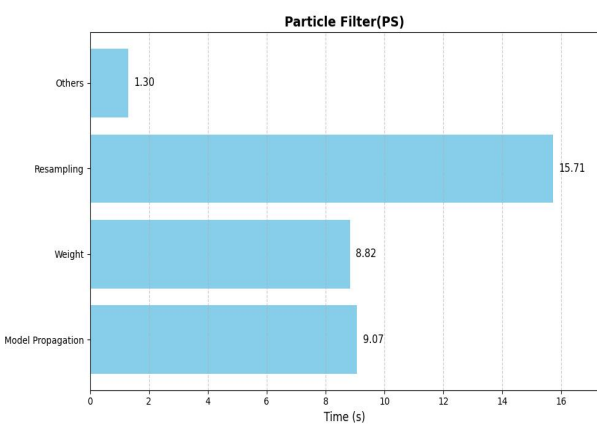


그림 2. 파티클 필터의 수행시간 프로파일링
Fig. 2. Profiling of the execution time for the Particle Filter

이러한 이유로 실시간으로 고속 탄도 목표물을 정밀하게 추적하기 위해 많은 입자 수를 활용하는

파티클 필터를 적용하기 어렵다는 한계가 존재했다. 따라서 파티클 필터의 성능 저하 없이 실시간성 개선을 위한 IP(Intellectual Property) 설계를 통해 많은 입자 수를 사용하면서 가속화 및 병렬화 기법을 적용하는 것이 필요하다. 본 논문에서는 수행시간이 가장 큰 리샘플링과 다음으로 큰 모델 전파 연산에 대해 병렬화 기법을 적용한 IP를 설계하여 복잡한 행렬 곱셈과 정렬 과정을 가속화하고 실시간성을 향상시킨다.

파티클 필터에 양자화를 적용한 대부분의 관련 연구는 양자화된 데이터를 파티클 필터를 적용하여 추정 정확도를 높이기 위해 연구가 진행되었다 [19]-[21]. 또한 파티클 필터의 가속화를 고려한 연구들은 대부분 실수 데이터를 사용하여 하드웨어 가속기를 설계한 연구들이다[16][23]. 이러한 연구들은 본 논문의 배경인 고속 탄도 목표물 추적에서 사용하고자 하는 저전력 AI 프로세서에 직접적으로 적용하기 어려우며 실시간성을 보장하기 어렵다. 따라서 제안하는 방법에서는 파티클 필터 연산에 사용되는 데이터를 양자화하고, 병목 구간을 IP로 설계하여 수행시간을 가속함으로써 저전력 AI 프로세서가 지원하는 정수형 기반의 연산 커널 작성과 실시간성을 보장하고자 한다.

III. 제안하는 기법

3.1 전체 구조

그림 3은 제안하는 양자화된 파티클 필터의 가속화 설계 구조이다. 많은 수행시간이 소요되는 리샘플링과 모델 전파를 PL(Programmable Logic)에서 병렬 처리를 진행하고, 전체적인 파티클 필터의 연산 및 제어는 PS(Processing System)에서 진행된다. 또한 PS와 PL 사이의 AXI(Advanced eXtensible Interface) 프로토콜을 사용한 스트림 데이터 전송을 통해 데이터 전송의 오버헤드도 최소화되도록 했다.

파티클 필터가 시작되면, 추정하려는 상태 공간에 파티클들을 무작위로 생성하고 초기화한다. 이후 현재 파티클들의 상태를 예측 모델에 따라 다음 시간 단계의 상태로 예측한다. 이 과정은 수많은 파티클에 대해 동일한 행렬 연산을 반복하므로 병렬 처

리에 매우 적합하다. 따라서 행렬 연산을 수행하는 IP를 설계하여 IP에서 사용되는 데이터를 PS에서 PL로 전송한다. 행렬 곱셈 IP에서는 정수형 기반의 행렬 곱 연산을 진행하기 위해 양자화, 행렬 곱, 역양자화를 순차적으로 처리하고 최종적으로 64비트의 예측 상태가 도출한다. PL에서 전달받은 예측된 파티클들이 실제 관측값과 얼마나 일치하는지 평가하여 각 파티클에 가중치를 계산한다. 이후 연산인 리샘플링은 가중치가 낮은, 즉 실제 관측과 거리가 먼 파티클은 제거하고, 가중치가 높은 파티클은 복제하여 새로운 파티클 집합을 만든다. 리샘플링에서 양자화를 진행하는 것은 결국 가장 높은 가중치를 추출하는 과정이기 때문에 역양자화 과정이 필요하지 않다. 따라서 리샘플링 IP에서는 양자화 이후 정렬, 인덱스 추출로 구성되어 있다. 재추출된 파티클 집합을 기반으로 최종 상태를 추정하고, 시스템 상태를 갱신한다.

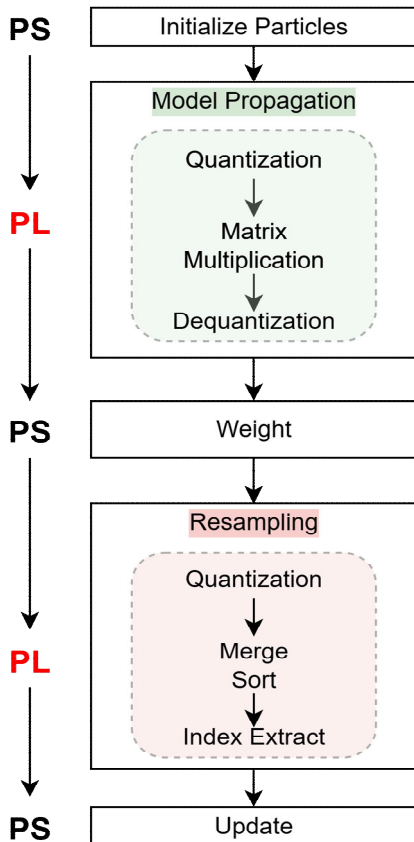


그림 3. 양자화된 파티클 필터의 가속화 설계 구조
Fig. 3. Architecture of an accelerated quantized Particle Filter

3.2 클리핑 기반 대칭 양자화 및 역양자화

상태 추정의 정확도를 위해 실수형 64비트의 데이터를 사용하여 정밀한 추정을 도출했다. 하지만 자원의 최적화, 저전력 AI 프로세서와의 호환성을 위해 정수형 기반의 데이터를 활용한 IP와 커널 설계가 필요하다. 따라서 실수형을 정수형으로 양자화를 진행하면서도 기존과 동일한 성능을 도출할 수 있도록 양자화 오차를 최소화하는 양자화 기법이 필요하다. 또한 정수형 연산 이후에 기존 실수형의 연산이 그대로 진행될 수 있도록 역양자화 과정을 통해서 다시 실수형으로 변환하는 과정이 필요하다.

본 논문에서는 최대 절댓값 기반의 스케일링과 클리핑 기법을 활용한 양자화 및 역양자화 기법을 적용했으며, 아래의 수식과 같다.

$$S_{x_p} = \frac{\max(|x_p|)}{2^{16} - 1} \quad (11)$$

$$x_{p_{quant}} = clip\left(\text{round}\left(\frac{M}{S_M}, -2^{16}, 2^{16} - 1\right)\right) \quad (12)$$

$$x_{p_{dequant}} = x_{p_{quant}} \cdot S_{x_p} \quad (13)$$

식 (11)을 통해 각 행렬의 최대 절댓값을 결정된 후, 배율 계수를 도출한다. 각 행렬의 원소 중 최대 절댓값을 찾고 이를 기반으로 데이터가 INT16 범위에 매핑되도록 배율 계수를 도출한다. 다음으로 식 (12)의 양자화 함수를 사용하여 각 행렬 요소를 개별적으로 양자화하고 값을 배정밀도에서 INT16으로 변환한다. 특히 양자화 함수에서는 배율 계수로 나눈 후 반올림 함수를 사용하여 모든 원소를 정수형으로 변환한다. 만약 정수형으로 변환한 결과가 INT16 범위를 벗어난다면 클리핑하여 값의 오버플로우를 방지한다. 이렇게 양자화된 행렬들을 모델 전과, 리샘플링에 필요한 수식에 적용하여 연산을 진행하고 도출된 결과를 식 (13)을 통해 다시 실수형으로 변환하는 역양자화 과정을 진행하여 이후의 연산이 그대로 유지되도록 한다.

그림 4는 목표물의 상태를 나타내는 x_p 행렬에 클리핑 기반의 양자화 기법을 적용하여 각 원소를

배정밀도에서 정수형으로 변환하는 과정을 보여준다. 각 원소에 대해 위의 수식을 순차적으로 적용하여 16비트의 범위로 매핑한다. 이후 변환된 데이터에 역양자화 과정을 각 원소에 적용하여 다시 배정밀도 64비트로 변환한다. 이를 통해 양자화 오차를 최소화하고, 정수화된 행렬을 기반으로 행렬 곱셈 연산을 진행할 수 있음을 확인할 수 있다.

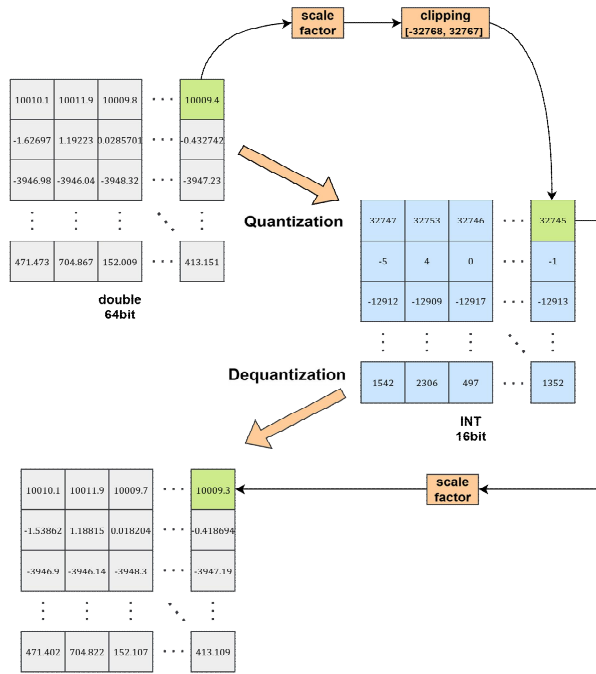


그림 4. 상태 행렬 x_p 에 적용한 양자화 기법

Fig. 4. Quantization of state matrix x_p

3.3 모델 전파 가속화 및 양자화 적용

모델 전파는 삼중 루프 기반의 행렬 곱셈을 포함하며 양자화 및 역양자화 단계가 추가되어 연산 복잡도가 높다. 따라서 양자화, 행렬 곱셈, 역양자화를 하나로 통합하고 효율적인 병렬 처리를 위해 최적화된 IP 블록을 설계했다. 표 1에서 본 논문에서 제안하는 모델 전파 IP의 구성을 확인할 수 있다. 높은 연산 복잡도로 인해 많은 자원을 사용하는 행렬 곱셈 연산을 정수형 기반의 행렬 곱셈으로 경량화하여 자원 사용량을 낮추고 계산 속도를 높일 수 있도록 클리핑 기반의 대칭 양자화 기법을 구현했다. 먼저 연산에 사용되는 각 행렬의 최대 절댓값을 찾고 이를 활용해서 배울 계수를 구한다. 배울 계수

는 실제 배정밀도 범위에서 얼마의 크기를 갖는지 나타내는 정밀도 값이다. 이후 행렬의 각 원소를 배울 계수로 나눈 후 반올림 및 임계값을 적용하여 INT16의 범위인 -32768과 32767 사이로 보정한다. 이 과정을 통해 원본 배정밀도 데이터의 분포 특성을 유지하면서 연산 속도 및 자원 효율에 유리한 정수를 사용하여 연산을 진행할 수 있다.

표 1. 모델 전파 IP 연산 구성

Table 1. Model propagation IP operation configuration

Input : x_p (Initial particle), A (State transition matrix), $\sqrt{Q_k}$ (Covariance matrix), $rand$ (random)
 output : x_p (Predicted target state)

1: Find maximum values & compute scale factors

$$S_{x_p} = \frac{\max(|x_p|)}{2^{16} - 1}$$

2: Quantize to INT16

$$x_{p_{quant}} = clip\left(\text{round}\left(\frac{M}{S_M}, -2^{16}, 2^{16} - 1\right)\right)$$

3: Matrix multiplication based on integer data :

for $c \leftarrow 0$ to $N-1$ do

for $i \leftarrow 0$ to $L-1$ do

$$temp[i] \leftarrow \sum_{j=0}^{L-1} (Q_{\sqrt{Q_k}}[i][j] \cdot Q_r[j][c])$$

$$y1[i] \leftarrow \sum_{j=0}^{L-1} (Q_A[i][j] \cdot Q_{x_p}[j][c])$$

$$y2[i] \leftarrow \sum_{j=0}^{L-1} (Q_A[i][j] \cdot temp[j])$$

end for

4: Dequantization

for $i \leftarrow 0$ to $L-1$ do

$$x_p[i][c] = y1[i] \cdot (S_A \cdot S_{x_p} + y2[i] \cdot (S_A \cdot S_{\sqrt{Q_k}} \cdot S_r))$$

end for

end for

그림 5는 양자화된 데이터를 이용해서 모델 전파 연산을 진행하는 과정을 보여준다. 양자화된 정수 행렬을 이용해 양자화된 $quant_A$ 와 $quant_x_p$ 를 곱하여 y_1 을 완성하고, 양자화된 $quant_A$, $quant_Q_k$, $quant_rand$ 를 곱하여 y_2 를 완성한다. 여기서 16비트 정수끼리 곱하면 결과값이 16비트 범위를 넘어갈 수 있으므로 데이터 손실을 막기 위해 더 넓은 범위인 32비트 정수 (INT32)에 중간 결과를 저장한다. 이후 y_1 과 y_2 에 배울 계수를 곱하여 기존의 배정밀

도로 복원된 두 값을 더해서 최종 상태 예측 결과를 도출한다.

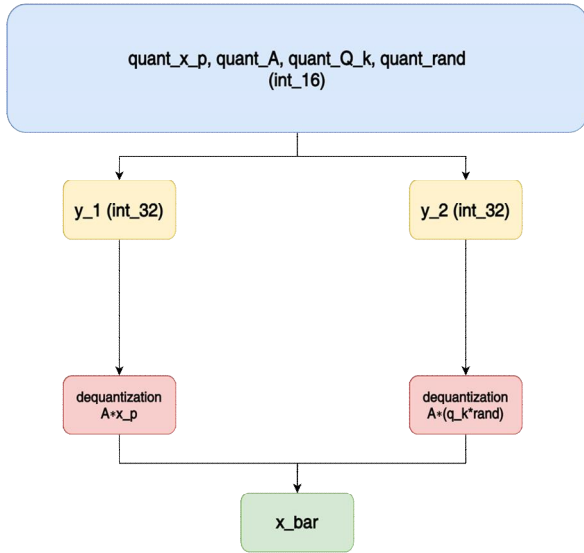
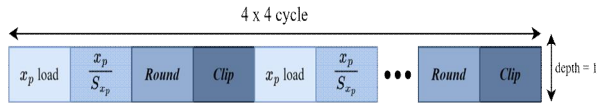
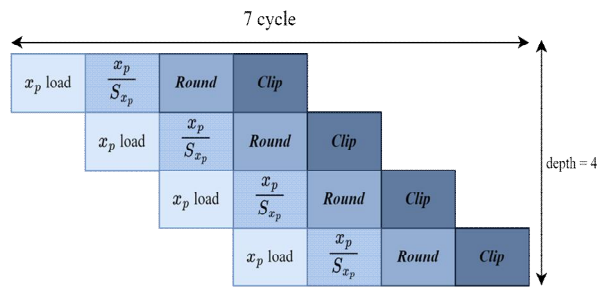


그림 5. 양자화 기법이 적용된 모델 전파 연산 과정
Fig. 5. Quantized model propagation process



(a) 순차적인 양자화 연산 과정
(a) Sequential quantization operation



(b) 파이프라인을 적용한 양자화 연산 과정
(b) Pipelined quantization operation

그림 6. 파이프라이닝 기법이 적용된 양자화 연산 구조
Fig. 6. Pipelined architecture for quantization operations

그림 6은 식 (11) ~ 식 (13)에서 진행되는 양자화 과정을 파이프라이닝 기법을 통해 가속화하여 전체 모델 전파 연산의 수행시간을 단축하기 위한 과정을 나타낸다. 그림 6(a)과 같이 순차적 연산에 비해 그림 6(b)는 파이프라이닝 기법을 적용하여 전체 사이클을 7사이클로 단축된 것을 확인할 수 있다. 이를

통해 모델 전파 연산에 포함되는 양자화 과정을 가속화하여 전체 수행시간을 효과적으로 단축시켰다.

3.4 리샘플링 가속화 및 양자화 적용

리샘플링은 가중치 연산을 통해 얻은 각 입자에 대한 가중치 값과 난수값을 합하여 정렬을 진행한다. 정렬을 진행하는 개수는 파티클의 2배이며, 이에 따라 많은 수행시간이 소요된다. 따라서 해당 연산을 양자화하여 경량화된 연산으로 변환하고 높은 가중치의 인덱스를 선택하여 복제를 진행한다. 이를 통해 높은 가중치를 가진 입자가 많이 선택될 수 있도록 한다. 여기서 인덱스만 추출하면 되기 때문에 모델 전파와는 다르게 역양자화를 진행할 필요가 없다. 따라서 클리핑 기반의 대칭 양자화, 정렬, 인덱스 추출에 대해 병렬화 기법을 적용하여 최적화된 IP 블록을 설계했으며 구성은 표 2에서 확인할 수 있다.

표 2. 리샘플링 IP 구성

Table 2. Resampling IP configuration

Input : ind1 array(weight, random)
output : x_p (Update Target State)

1: Quantize to INT16

$$Array_{quant} = clip\left(round\left(\frac{Array}{S_{Array}}, -2^{16}, 2^{16} - 1\right)\right)$$

2: Initialize sub array size

3: Merging two adjacent subarrays

4: Compare left and right element

5: If left element \leq right element :

copy left element & its index to temporary storage

Else right element < left element :

copy right element & its index to temporary storage

6: If one group is fully merged:

Copy all remaining elements and indices from the other

group to temporary storage

Copy merged results from temporary storage back into the original arrays (values and indices)

7: Compute resampled index

리샘플링은 정렬 연산으로 인해 많은 종속성이 존재하기 때문에 루프 언롤링과 같은 한 사이클에 전부 처리하는 기법보다는 파이프라이닝과 같이 순

서는 존재하지만 여러 단계로 나눠서 동시에 단계별로 처리하는 기법이 적절하다. 한 클럭 사이클에서 가중치와 난수값이 병합된 배열 $ind1[i-1]$ 을 메모리에서 읽어오는 동안 이전 클럭 사이클에 읽어온 $ind1[i-1]$ 값은 최댓값과 비교되고, 그 이전의 $ind1[i-2]$ 값은 양자화가 수행되는 구조다. 여러 연산이 겹쳐서 동시에 실행되므로 순차적으로 실행할 때보다 빠르게 데이터 준비를 마칠 수 있다. 연산량이 많은 정렬과 데이터 복사 과정에서 파이프라이닝의 효과가 극대화됐다.

IV. 결 과

4.1 환경 설정

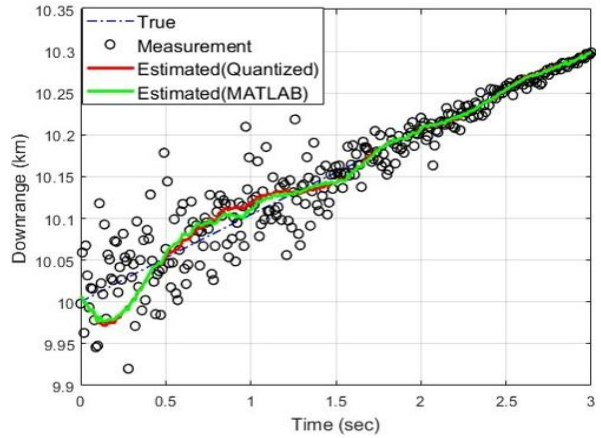
본 논문은 듀얼 ARM Cortex-A9 코어 프로세서 기반의 Xilinx Zynq ZC706 Evaluation Kit 보드를 통해 개발을 진행했다. 제안된 기법의 효과는 탄도 표적 추적 시뮬레이션을 통해 입증한다. 시뮬레이션 시간은 3초, 시간 간격은 0.01초로 설정되어 총 200 단계의 시간 스텝을 갖는다. 동적 모델은 식 (1)부터 (6)을 사용하여 정의되며 미사일의 공기역학적 항력과 무게 사양은 [24]에 설정된 바와 같이 실제 궤적과 유사하게 C_w, β 의 항력, 탄도 계수를 설정했다. 레이더 수신 잡음(n_ψ, n_θ, n_R)은 가우시안 혼합 분포를 따르도록 각각 $0.1^\circ, 0.1^\circ, 1m$ 로 설정했다. 레이더는 지상에 고정되어 있으며 표적 추적을 위해 싱어 모델을 사용하며, 2,000개의 파티클을 사용한다.

4.2 결과 및 분석

그림 7에서 점선은 True 값, 원은 측정값, 초록색 선은 배경밀도를 사용하여 추정한 시뮬레이션 결과, 빨간색 선은 양자화 및 병렬화 기법을 적용하여 추정한 시뮬레이션 결과를 나타낸다.

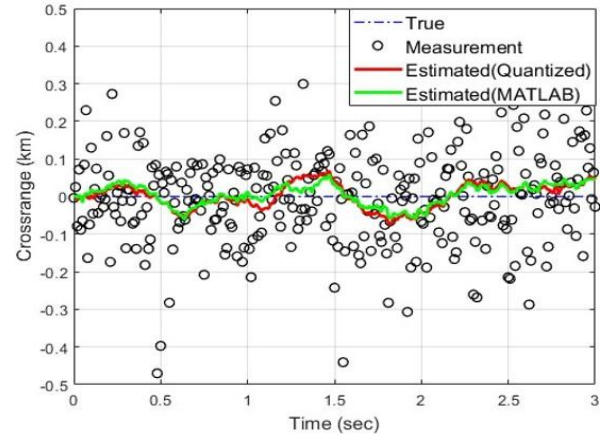
측정값에 잡음이 존재하여 불안정함에도 배경밀도 연산과 양자화된 INT16 연산 모두 안정적으로 목표물을 추정하는 것을 확인할 수 있다. 양자화 기법을 적용함으로써 약간의 양자화 오차는 존재하지

만, 두 추정 그래프의 차이가 크지 않고 최종적으로 동일한 성능의 상태 추정이 가능함을 보여준다.



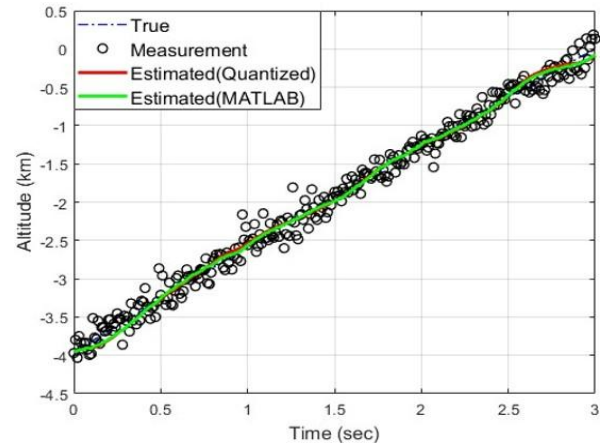
(a) 추정된 표적의 수평 거리

(a) Estimated target downrange



(b) 추정된 표적의 측면 거리

(b) Estimated target cross range



(c) 추정된 표적의 고도

(c) Estimated target altitude

그림 7. 파티클 필터를 이용한 타겟 위치 추정 결과
Fig. 7. Target position estimation results using Particle Filter

그림 8은 기존의 파티클 필터와 양자화 기법을 적용한 파티클 필터의 수행시간을 비교한다. PS 수행시간을 비교하면, 클리핑 기반의 대칭 양자화의 수행시간이 추가되면서 전체 수행시간이 증가한 것을 확인할 수 있다. 하지만 병렬화 기법을 적용한 PS&PL의 수행시간을 비교하면 기존 파티클 필터는 1.53배 단축되었고 제안하는 방법은 1.93배 수행시간 단축이 이뤄졌다. 이는 기존 파티클 필터의 가속화 결과 대비 13.3% 개선된 결과이다.

동일한 기법이 같은 연산에 적용됐음에도 불구하고, 정수형 기반의 행렬 곱셈이 수행되기 때문에 데이터의 경량화로 수행시간이 단축된 것을 확인할 수 있다. 또한 정수형 기반의 데이터를 사용하면서 자원 사용량이 줄어들어 병렬화 기법에 사용할 수 있는 자원이 증가하면서 파이프라이닝의 간격을 더 촘촘하게 설계하여 수행시간이 줄어든 것을 볼 수 있다.

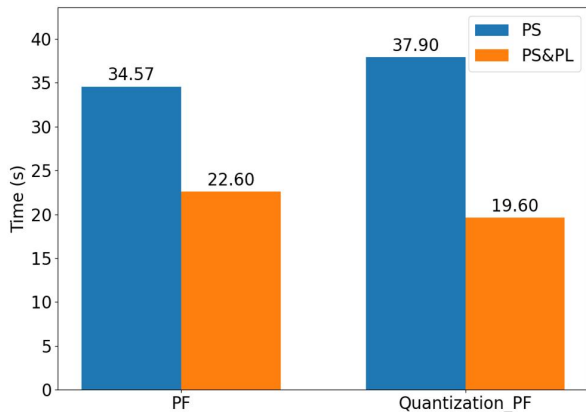


그림 8. 양자화 기법이 적용된 파티클 필터와 기존 파티클 필터의 수행시간 결과 비교
 Fig. 8. Computation time comparison of a quantized Particle Filter and Particle Filter

그림 9는 양자화된 파티클 필터의 PS 수행시간과 병렬화 기법이 적용된 PS&PL 수행시간을 비교한다. 가속화를 위해 IP를 설계하고 이를 적용한 모델 전파와 리샘플링 연산에 대해 수행시간이 각각 72%, 61% 단축되었다. 또한 기존의 병목 연산들이 가속화되면서 전체 수행시간 역시 48% 단축되었다. 이는 연산의 특성을 고려한 하드웨어와 소프트웨어의 공동 설계 결과가 효율적임을 나타내며 각 연산 특성에 맞게 병렬화 기법을 적절하게 적용했음을 보여준다.

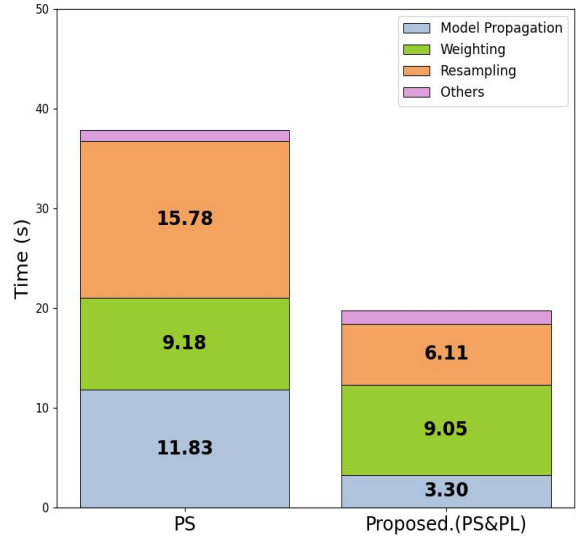


그림 9. 양자화된 파티클 필터의 연산 수행시간 비교
 Fig. 9. Comparing the computational time of a quantized Particle Filter

표 3. FPGA 자원 사용량

Table 3. FPGA resource utilization

Utilization	PF	Quantized PF
LUT	8%	15%
LUTRAM	1%	2%
FF	5%	8%
BRAM	40%	40%
DSP	15%	10%
BUFG	3%	3%

표 3은 제안하는 기법의 하드웨어 자원 사용량을 나타낸다. 이를 통해 클리핑 기반의 대칭 양자화 기법을 적용함으로써 정수형의 연산이 가능한 Quantized PF에서 DSP 사용량이 줄어든 것을 확인할 수 있으며, 그로 인해 더 많은 자원 사용이 가능하므로 간격을 더 촘촘하게 설계하여 LUT의 사용량이 소폭 증가한 것을 확인할 수 있다. 또한 고차원의 상태 행렬을 사용하기 때문에 BRAM의 사용량이 가장 높은 사용량을 차지하며, 이는 향후 데이터 전송 시 데이터를 분할하여 반복적으로 전송함으로써 줄일 수 있는 개선의 여지가 존재한다.

V. 결 론

본 연구에서는 고속 탄도 목표물 추적과 같이 비선형 및 비가우시안 특성을 갖는 상태 추정 문제를

해결하기 위해 파티클 필터를 사용했다. 기존의 파티클 필터는 많은 파티클을 순차적으로 처리하는 것이 연산 시간을 크게 증가시키기 때문에 이러한 시나리오에서 종종 비효율적이다. 따라서 연산 과정을 가속하면서 최신 저전력 AI 프로세서와의 호환성을 가질 수 있는 양자화된 병렬 파티클 필터링을 제안한다. 또한 Zynq 계열의 FPGA 보드를 활용하여 병목 현상을 일으키는 연산에 대한 IP를 설계하여 병렬화 기법을 적용한 하드웨어 구조를 제안한다. 배정밀도 행렬 대신 INT16으로 변환하기 위해 클리핑 기반의 대칭 양자화를 적용한 정수형 기반의 행렬을 사용하고 전체 연산 과정에 파이프라이닝과 언롤링을 적용함으로써 제안된 방법은 PS 수행시간과 비교하여 48%의 수행시간 단축을 달성했다. 특히 모델 전파 연산과 리샘플링은 PS 단독 연산과 비교하여 실행 시간에서 72%, 61%의 수행시간 단축이 이뤄졌다. 측정 잡음에도 불구하고 양자화된 파티클 필터는 실제 표적값을 정확하게 추적했으며, 이를 통해 실시간 고속 탄도 목표물 추적에 대한 적합성을 입증했다. 또한 클리핑 기반의 대칭 양자화 과정이 추가적으로 수행됨에 따라 일부 양자화 오차가 발생했지만 역양자화를 통해 기존의 배정밀도 기반의 연산이 문제없이 진행될 수 있게 함으로써 상태 추정에는 영향이 미미하여 기존과 동일한 수준의 성능 결과를 도출하는 것을 확인했다. 향후 연구에서는 제안된 방법을 AMD Versal과 같은 최신 저전력 AI 프로세서에 구현함으로써 파티클 필터를 더욱 최적화하는 것을 목표로 연구를 수행할 예정이다.

References

- [1] P. Siting, Z. Qian, M. Yichao, and X. Cheng, "Observability Enhancement of the Target Tracking Model Based on Infrared Semi-strapdown Seeker", 2019 IEEE International Conference on Unmanned Systems (ICUS), Beijing, China, pp. 183-188, Oct. 2019. <https://doi.org/10.1109/ICUS48101.2019.8996025>.
- [2] L. Baomin and P. Xiefan, "Research on Maneuvering Target Tracking Algorithm Based on an Improved Particle Filter Algorithm", 2024 OES China Ocean Acoustics (COA), Harbin, China, pp. 1-4, May 2024. <https://doi.org/10.1109/COA58979.2024.10723380>.
- [3] Y. Sun, J. Xie, and J. Guo, "A new maneuvering target tracking method using adaptive cubature Kalman filter", 2014 IEEE International Conference on Control Science and Systems Engineering, Yantai, China, pp. 40-44, Dec. 2014. <https://doi.org/10.1109/CCSSE.2014.7224505>.
- [4] W. Teng, "Radar Tracking Simulation Numerical Model Based on Improved Kalman Filter EKF Algorithm", 2023 5th International Conference on Artificial Intelligence and Computer Applications (ICAICA), Dalian, China, pp. 836-840, Nov. 2023. <https://doi.org/10.1109/ICAICA58456.2023.10405645>.
- [5] J. Cao, X. Zhang, H. Hao, and X. Shi, "Noncircular Signal Tracking With Distributed Passive Arrays: Combining Data Fusion and Extended Kalman Filter", IEEE Sensors Journal, Vol. 24, No. 1, pp. 757-768, Jan. 2024. <https://doi.org/10.1109/JSEN.2023.3333863>.
- [6] Y. Liu and Z. Sun, "EKF-Based Adaptive Sensor Scheduling for Target Tracking", 2008 International Symposium on Information Science and Engineering, Shanghai, China, pp. 171-174, Dec. 2008. <https://doi.org/10.1109/ISISE.2008.286>.
- [7] S. J. Yoon, N. Kim, H. Lee, I. Lim, G. Kwon, and J. Park, "Parallelized Huber-based EKF on FPGA for Real-Time High-Speed Target Tracking", The Journal of Korean Institute of Information Technology, Vol. 23, No. 3, pp. 77-87, Mar. 2025. <https://doi.org/10.14801/jkiit.2025.23.3.77>.
- [8] X. Liu and B. Mo, "Line-of-sight Estimation for Missile with Roll-Pitch Seeker", 2018 37th Chinese Control Conference (CCC), Wuhan, China, pp. 4957-4961, Jul. 2018. <https://doi.org/10.23919/ChiCC.2018.8482856>.
- [9] B. Jung, C.-S. Lee, and W.-S. Ra, "Ballistic

- Target Tracking Using Range Spread Measurements of a Wideband Radar Seeker", 2024 27th International Conference on Information Fusion (FUSION), Venice, Italy, pp. 1-7, Jul. 2024. <https://doi.org/10.23919/FUSION59988.2024.10706509>.
- [10] D. Kim, et al., "Accelerated Particle Filter With GPU for Real-Time Ballistic Target Tracking", *IEEE Access*, Vol. 11, pp. 12139-12149, Jan. 2023. <https://doi.org/10.1109/ACCESS.2023.3238873>.
- [11] L. Fan, B. Hongkui, X. Jiajun, Y. Chenlong, and L. Xuhui, "A dual channel perturbation particle filter algorithm based on GPU acceleration", *Journal of Systems Engineering and Electronics*, Vol. 29, No. 4, pp. 854-863, 2018. <https://doi.org/10.21629/jsee.2018.04.20>.
- [12] V. P. Jilkov, J. Wu, and H. Chen, "Performance comparison of GPU-accelerated particle flow and particle filters", *Proc. of the 16th International Conference on Information Fusion*, Istanbul, Turkey, pp. 1095-1102, Jul. 2013.
- [13] D. Kim, H. Lee, H.-H. Kwon, Y. Hwang, and W. Choi, "Parallelized Particle Filter With Efficient Pipelining on FPGA for Real-Time Ballistic Target Tracking", *IEEE Access*, Vol. 11, pp. 104830-104838, Sep. 2023. <https://doi.org/10.1109/access.2023.3317896>.
- [14] A. Jarrah, M.-M. Jamali, and S.-S.-S. Hosseini, "Optimized FPGA based implementation of particle filter for tracking applications", *NAECON 2014 IEEE National Aerospace and Electronics Conference*, Dayton, OH, USA, pp. 233-236, Jun. 2014. <https://doi.org/10.1109/naecon.2014.7045808>.
- [15] S.-A. Li, C.-C. Hsu, W.-L. Lin, and J.-P. Wang, "Hardware/software co-design of particle filter and its application in object tracking", *Proc. 2011 International Conference on System Science and Engineering*, Macau, China, pp. 87-91, Jun. 2011. <https://doi.org/10.1109/icsse.2011.5961879>.
- [16] Y. Imamura, S. Fukui, T. Manabe, and Y. Shibata, "An FPGA-based Power-saving Particle Filter Using Dynamic Reconfiguration", 2021 Ninth International Symposium on Computing and Networking Workshops (CANDARW), Matsue, Japan, pp. 439-443, Nov. 2021. <https://doi.org/10.1109/candarw53999.2021.00081>.
- [17] M. Y. Lee, J. Chung, J. H. Han, and Y. S. Kwon, "Trends in AI Processor Technology", *Electronics and Telecommunications Trends*, Vol. 35, No. 3, pp. 66-75, Jun. 2020. <https://doi.org/10.22648/ETRI.2020.J.350307>.
- [18] Y. Liu, Z. Wang, C. Liu, M. Coombes, and W.-H. Chen, "A Novel Algorithm for Quantized Particle Filtering With Multiple Degrading Sensors: Degradation Estimation and Target Tracking", *IEEE Transactions on Industrial Informatics*, Vol. 19, No. 4, pp. 5830-5838, Apr. 2023. <https://doi.org/10.1109/TII.2022.3176910>.
- [19] S. Hong, M. Bolic, and P. M. Djuric, "An efficient fixed-point implementation of residual resampling scheme for high-speed particle filters", *IEEE Signal Processing Letters*, Vol. 11, No. 5, pp. 482-485, May 2004. <https://doi.org/10.1109/LSP.2004.826634>.
- [20] Y. Wang, X. Wang, Y. Shan, and N. Cui, "Quantized genetic resampling particle filtering for vision-based ground moving target tracking", *Aerospace Science and Technology*, Vol. 103, pp. 105925, Aug. 2020. <https://doi.org/10.1016/j.ast.2020.105925>.
- [21] R. Karlsson and F. Gustafsson, "Particle filtering for quantized sensor information", 2005 13th European Signal Processing Conference, Antalya, Turkey, pp. 1-4, Sep. 2005.
- [22] Y. Ruan, P. Willett, A. M[ar]s, F. Palmieri, and S. Marano, "Practical fusion of quantized measurements via particle filtering", *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 44, No. 1, pp. 15-29, Jan. 2008. <https://doi.org/10.1109/TAES.2008.4516986>.

[23] C.-C. Hsu, W.-C. Kao, Y.-C. Chu, S.-A. Li and W.-L. Lin, "Object tracking based on hardware/software co-design of particle filter and particle swarm optimization", 2014 IEEE Fourth International Conference on Consumer Electronics Berlin (ICCE-Berlin), Berlin, Germany, pp. 225-227, Sep. 2014. <https://doi.org/10.1109/ICCE-Berlin.2014.7034287>.

[24] D. C. Wright and T. Kadyshchev, "An analysis of the north Korean nodong missile", Science & Global Security, Vol. 4, No. 2, pp. 129-160, 1994. <https://doi.org/10.1080/08929889408426397>.

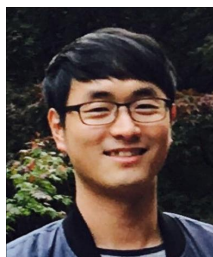
저자소개

김 나 연 (Nayeon Kim)



2024년 2월 : 금오공과대학교
전자공학부(공학사)
2024년 3월 ~ 현재 :
금오공과대학교 IT융복합공학과
석사과정
관심분야 : SLAM, Algorithm
acceleration with GPU and
FPGA, embedded system

이 현 철 (Heoncheol Lee)



2006년 8월 : 경북대학교
전자전기컴퓨터학부(공학사)
2008년 8월 : 서울대학교
전기컴퓨터공학과(공학석사)
2013년 8월 : 서울대학교
전기컴퓨터공학과(공학박사)
2013년 9월 ~ 2019년 2월 :

국방과학연구소 선임연구원
2019년 3월 ~ 2023년 8월 : 금오공과대학교 전자공학부
IT융복합공학과 조교수
2023년 9월 ~ 현재 : 금오공과대학교 전자공학부
IT융복합공학과 부교수
관심분야 : SLAM, 자율주행, 인공지능, 알고리즘 가속화

권 혁 훈 (Hyuckhoon Kwon)



2002년 2월 : 한국과학기술원
항공우주공학과(공학사)
2005년 2월 : 한국과학기술원
항공우주공학과(공학석사)
2020년 2월 : 한국과학기술원
항공우주공학과(공학박사)
2005년 3월 ~ 2009년 1월 :

현대자동차 연구원
2009년 2월 ~ 현재 : LIG넥스원 수석연구원
관심분야 : 컨벡스 최적화, 최적제어, 유도조종
알고리즘, 비선형 제어, 머신 러닝

최 원 석 (Wonseok Choi)



2006년 2월 : 홍익대학교
전기전자공학부(공학사)
2017년 2월 : 연세대학교
국방공학과(공학석사)
2007년 3월 ~ 현재 : LIG넥스원
수석연구원
관심분야 : FPGA, 필터, 센서

방위산업

정 보 라 (Bora Jung)



2020년 2월 : 한양대학교
전기전자공학부(공학사)
2020년 3월 ~ 현재 : LIG넥스원
선임연구원
관심분야 : FPGA, 필터, 센서
방위산업