



Received: 2023/07/20
Revised: 2023/08/04
Accepted: 2023/08/23
Published: 2023/09/30

***Corresponding Author:**

Sunghyun Park

PGM RF & IIR Seeker R&D Lab, LIG Nex1

207, Mabuk-ro, Giheung-gu, Yongin-si, Gyeonggi-do,
Republic of Korea

E-mail: sunghyun.park@lignex1.com

Abstract

본 논문에서는 영상에서의 mean filter의 효과 및 영향을 확인하였다. 먼저, 일반적인 kernel에 대한 mean filter의 효과 및 처리속도에 대해 확인하였다. 해석 결과, 이론과 동일한 결과가 나왔으나 kernel size에 따라 kernel 내 픽셀 연산 및 영상지연을 확인할 수 있었다. 더불어 영상지연을 최소화하기 위한 방식을 제안하고 실제 영상에 대해 기존 mean filter와 차이를 확인하였다.

In this study, we would like to confirm the effect of the Mean Filter in image process. First, we made a test about the effect of mean filter on general Kernel and the processing time. As a result of analysis, the same result as the theory was obtained, but result of image has different pixel calculation and image delay according to the Kernel size. Additionally, a method for minimizing image delay was studied and differences from mean filter for actual images.

Keywords

열상 영상(Thermal Image),
적외선(Infrared Rays),
영상 처리(Image Processing),
평균 필터(Mean Filter)

Acknowledgement

이 논문은 2023년 정부(방위사업청)의 재원으로 수행된 연구임

이 논문은 2023년도 한국해군과학기술학회 하계학술대회 발표 논문임.

효율적인 Thermal Noise Filter 설계 및 분석

Efficient Design and Analysis of Thermal Noise Filter

박성현^{1*}, 박진호², 김홍락², 유연덕³, 강지안⁴

¹LIG넥스원 PGM 탐색기연구소 선임연구원

²LIG넥스원 PGM 탐색기연구소 수석연구원

³LIG넥스원 PGM 탐색기연구소 연구원

⁴국방과학연구소 미사일연구실 선임연구원

Sunghyun Park^{1*}, Jin-Ho Park², Hong-Rak Kim², Yeondeok Yoo³, Jian Kang⁴

¹Research engineer, PGM RF & IIR Seeker R&D Lab, LIG Nex1

²Chief research engineer, PGM RF & IIR Seeker R&D Lab, LIG Nex1

³Research engineer, PGM RF & IIR Seeker R&D Lab, LIG Nex1

⁴Research engineer, Missile R&D Lab, Agency for Defense Development

1. 서론

절대온도 0 K 이상의 모든 물체는 적외선 에너지를 방사한다. 방사된 적외선 에너지는 렌즈 및 광학계를 통과하여 검출기에 의해 전기적 신호로 변환된다. 물체가 발산하는 적외선 에너지를 탐지하여 영상으로 전시하는 장비를 열상 장비라고 한다[1].

열상 영상 에너지는 대부분 표적 에너지 그리고 내부온도에 의한 에너지로 구성된다. 적외선 검출기는 열에 의한 영향에 민감하며 특히 고온 등에 오래 노출되면 열의 영향에 민감하여 영상에 열잡음에 의한 에너지가 존재한다. 이러한 잡음을 보정하기 위해 널리 사용되는 방법으로 불균일 보정 기법(non-uniformity correction)이 있다. 열상 장비를 제작하는 회사들은 각 제품의 특성에 맞는 불균일 보정 기법을 적용하고 있다. 하지만 불균일 보정 기법은 출력이 고르지 않은 검출기 픽셀에 대해 출력 개형을 보정할 수는 있으나 실시간으로 변하는 외부 환경에 의해 발생하는 잡음 및 노이즈를 보정하기에는 한계가 있다[2].

본 논문에서는 불균일 보정된 영상을 mean filter에 통과시켜 영상을 분석하고 열잡음을 최소화하는 방법에 대해 제안하고자 한다. Mean filter의 kernel size에 따라 영상지연이 존재하는데 원래의 mean filter 계산 방식이 아닌 영상지연 최소화를 위한 방법을 적용하였다. 기존 계산 방식과 본 논문에서 적용된 계산 방식의 영상 및 지

연시간을 비교 분석하였으며 자체 영상 분석 툴을 통해 검증하였다.

2. 검출기 출력

감시·정찰분야에서는 주로 중적외선(5 μm - 8 μm)과 원적외선(8 μm - 12 μm) 파장대역의 검출기를 주로 사용한다. 물체에서 방사되는 적외선 에너지가 광학계와 검출기 FPA(focal plane array)를 통해 전기적인 신호로 변환된다. FPA는 적외선을 감지하여 발생한 광전류를 ROIC(readout integrated circuit)를 통해 전기적인 신호로 전달한다.

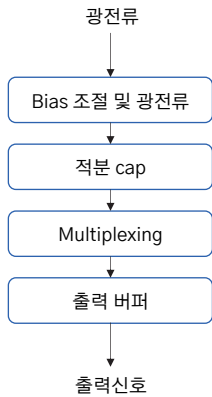


Fig. 1. 검출기 ROIC 동작

$$L_{sys} = L_{scene}(\lambda, T_{scene}) \cdot \Gamma_{optics} + L_{BB}(\lambda, T_{BB}) \quad (1)$$

검출기 출력신호는 여러 노이즈 성분을 포함하고 있다. 이러한 노이즈 성분 중에는 열에 의한 잡음 노이즈가 다수 포함되어 있다. 여러 보정과정을 통해 이를 처리함으로써 우리가 보기 편한 영상을 구현하게 된다. 이번 논문에서는 이러한 잡음을 제거하기 위한 방식을 제안하고자 한다.

3. Image padding을 통한 mean filter 설계

일반적으로 mean image는 기존 영상에 kernel size 만큼 image padding을 한 다음 convolution 연산을 통해 얻게 된다. N×M 크기의 이미지가 K×K 크기의 kernel과 convolution 연산하게 되면 출력 이미지 크기는 (N - K + 1) × (M - K + 1)으로 축소하게 되며 이를 식으로 표현하면 아래와 같다.

$$I_{out}(x, y) = \sum_i \sum_j I_{in}(x - i, y - i) \times K(i, j) \quad (2)$$

홀수의 kernel size 및 000 × 000 크기의 영상 데이터의 EDGE 값으로 padding 하였다. Kernel size에 따라 출력값은 달라지며 이에 따른 효과는 시뮬레이션을 통해 확인하였다.

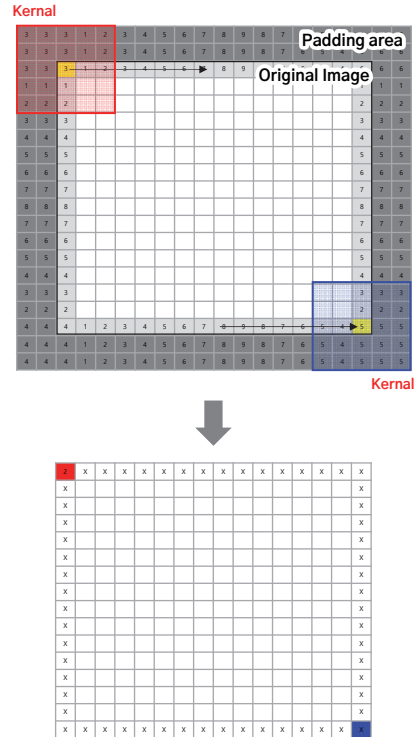


Fig. 2. Image padding (kernel size = 5)

4. 적분영상(integral image)을 이용한 평균 계산

Kernel 내 평균 연산을 쉽게 수행하는 방법으로 적분 영상(integral image)을 활용할 수 있다. 적분영상은 블록 내 값을 빨리 계산하고 물체의 특징을 추출하기 위한 과정으로 사용되었다[3]. 적분영상은 입력 영상의 내부 값의 합이나 kernel 내에 속한 영역의 합을 구하는데 유용하며 식 (3)과 같이 계산한다.

$$IntegImage(x, y) = \sum_{y=0}^y \sum_{x=0}^x orgImage(x', y') \quad (3)$$

Fig. 4에서 A, B, C, D의 integral image 값은 식 (4)와 같으며 박스 5×5의 합 D = D - B - C + A와 같다.

64	2	3	61	60	6	7	57
9	55	54	12	13	51	50	16
17	47	46	20	21	43	42	24
40	27	27	37	36	30	31	33
32	34	35	29	28	38	39	25
41	23	22	44	45	19	18	48
49	15	14	52	53	11	10	56
8	58	59	5	4	62	63	1

64	66	69	130	190	196	203	260
73	130	187	260	333	390	447	520
90	194	297	390	484	584	683	780
130	260	390	520	650	780	910	1040
162	326	491	650	808	976	1145	1300
203	390	577	780	983	1170	1357	1560
252	454	655	910	1166	1364	1561	1820
260	520	780	1040	1300	1560	1820	2080

Original image (input image)

Integral image

Fig. 3. 입력 영상과 적분영상의 예

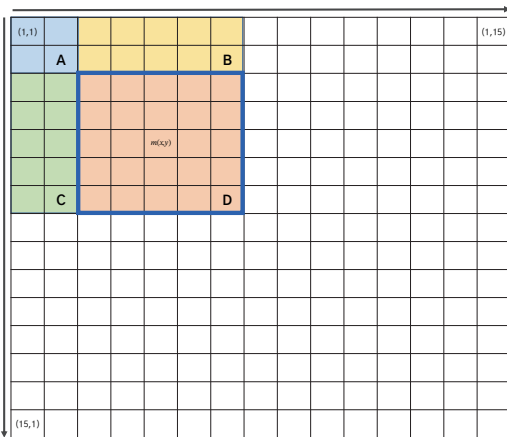


Fig. 4. 적분영상 영역 세분화

$$A = orgImage[(1,1) + (1,2) + (2,1) + (2,2)]$$

$$B = \sum_{x=1}^2 \sum_{y=1}^7 orgImage(x, y)$$

$$C = \sum_{x=1}^7 \sum_{y=1}^2 orgImage(x, y)$$

$$D = \sum_{x=1}^7 \sum_{y=1}^7 orgImage(x, y)$$

(4)

대부분 영상처리는 FPGA(field programmable gate array) 기반으로 처리되고 있다[4]. FPGA 기반 영상 처리는 실시간 연산을 위해 pixel 단위로 연산을 수행하며 적분영상을 활용한 mean 계산은 FPGA 기반으로 처리하기에는 많은 데이터양과 영상지연이 갖는다.

5. 데이터 최적화를 위한 평균 계산

이를 보완하기 위하여 가로 평균 연산 후 세로 평균 연산하는 방식을 적용하였다. 이를 식으로 표현하면

식 (5)와 같다.

$$\begin{aligned}
 Img_{row,avg} &= Img_{pad} \times Kernel_{1,K} \\
 Img_{tot,avg} &= Img_{row,avg} \times Kernel_{K,1}
 \end{aligned}
 \tag{5}$$

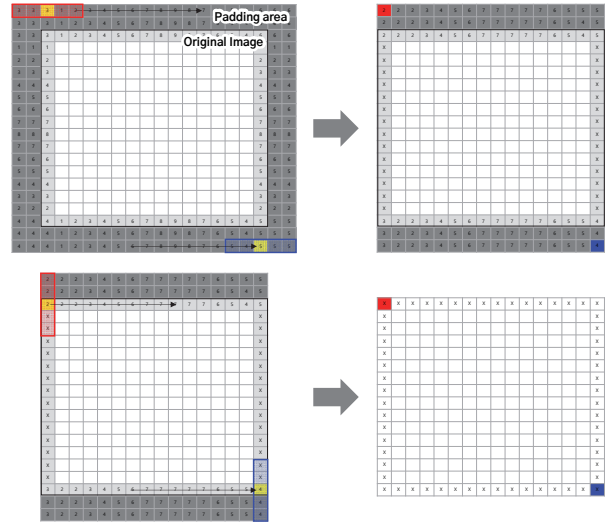


Fig. 5 가로 평균 연산 후 세로 평균 연산 방식

이 방식은 kenal size가 작으면 원래 평균값과 차이가 있지만 kenal size가 크면 그 차이를 최소화할 수 있으며 영상지연 및 데이터양을 줄일 수 있다.

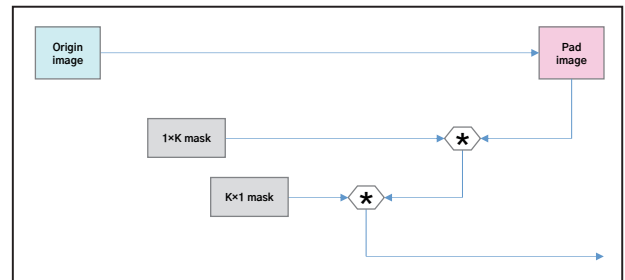


Fig. 6. 가로 평균 연산 후 세로 평균 연산의 플로우차트

6. Thermal noise filter 설계

Thermal noise filter는 입력 영상을 기반으로 열잡음 보정 filter의 효과를 확인하고자 한다. Thermal noise filter는 식 (6)과 같이 계산되며 Fig. 7과 같은 과정을 통해 영상처리를 한다.

$$Img_{TNF} = Img_{org} - Img_{Mean} + Mean_{min}
 \tag{6}$$

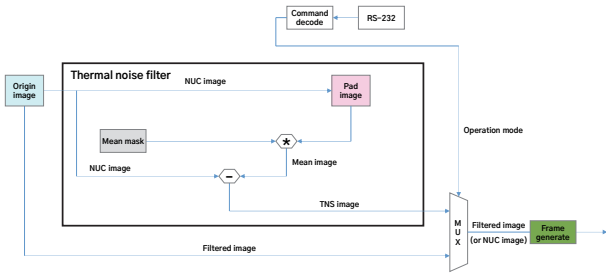


Fig. 7. Thermal noise filter의 영상 처리과정

최종 thermal noise filter의 값은 입력 데이터와 kernel size에 따라 음의 값이 나오는 경우가 있다. 이러한 경우에는 0으로 처리하였으며 평균의 최솟값은 한 프레임의 평균 최솟값으로 계산하였다.

6. 시뮬레이션

시뮬레이션을 위해 자체 개발한 소프트웨어를 활용하였으며 주차장의 000 × 000 크기의 14 bit 영상으로 분석을 진행하였다. Kernel size는 다양한 크기의 홀수값으로 분석을 진행하였다. 적분영상(integral image) 방식과 제안하는 방식의 영상 차이는 육안으로 확인하기엔 다소 어려움이 있어 제안하는 방식의 영상만 첨부하였다.



Fig. 8. 입력 영상(왼쪽) 및 히스토그램(오른쪽)

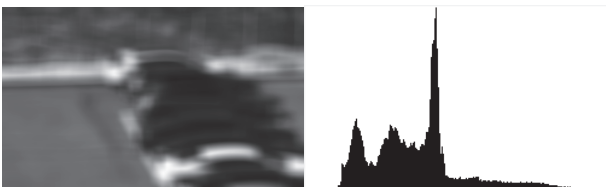


Fig. 9. Size 31 영상(왼쪽) 및 히스토그램(오른쪽)

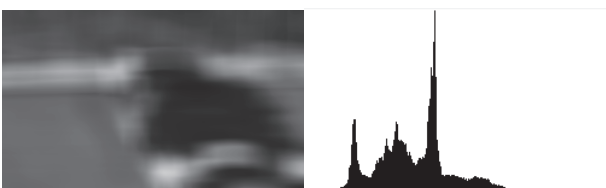


Fig. 10. Size 61 영상(왼쪽) 및 히스토그램(오른쪽)

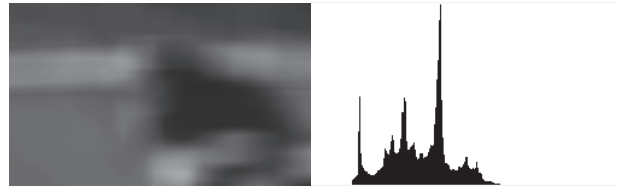


Fig. 11. Size 91 영상(왼쪽) 및 히스토그램(오른쪽)

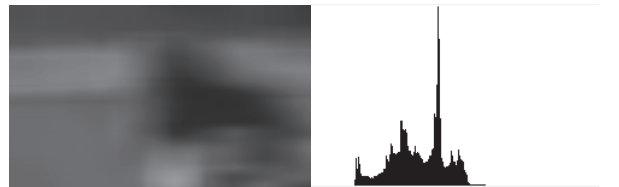


Fig. 12. Size 121 영상(왼쪽) 및 히스토그램(오른쪽)

Table 1. Mean 방식에 따른 영상 평균값(50회) 비교

Kernel size	Mean 방식	
	Integral image	제안하는 방식
0	6343.07	6343.07
31	6269.16	6268.70
61	6285.35	6284.87
91	6296.50	6296.01
121	6388.09	6387.60

Table 2. Mean 방식에 따른 연산횟수 비교

Kernel size	Mean 방식	
	Integral image	제안하는 방식
0	0	0
31	295,219,200	19,046,400
61	1,143,091,200	37,478,400
91	2,543,923,200	55,910,400
121	4,497,715,200	74,342,400

Simulation을 통해 kernel Size가 커질수록 영상은 흐릿해지는 것을 확인할 수 있다. 또한 히스토그램의 모양은 비슷하게 유지하면서 같은 값에 많이 분포하는 것을 확인할 수 있다.

Mean 방식에 따른 영상 평균값은 50회 프레임 평균값으로 비교하였으며 14 bit 영상에서 약 0.4 차이가 나는 것을 확인하였다.

연산횟수는 mean 연산을 위해 픽셀을 덧셈하는 연

산의 횡수를 카운트한 것으로 식 (7)과 같다.

$$\begin{aligned} & - \text{Integral image 방식: } K_{size}^2 \times 640 \times 480 \\ & - \text{제안하는 방식: } 2 \times K \times 640 \times 480 \end{aligned} \quad (7)$$

Kernal size가 커질수록 integral image 방식이 월등히 많아졌으며 소요시간도 늘어나는 것을 확인할 수 있었다. 연산횟수 및 시간을 고려했을 때 데이터가 큰 영상으로 mean 처리하는 상황에서는 제안하는 방식이 효율적이라고 판단된다.

7. 결론

본 논문에서는 mean filter 계산을 위한 방식에 대해 제시하였다. 대부분 integral image 방식의 kernal 내 평균을 계산하는 방식을 사용하지만 이는 FPGA 기반의 영상처리 환경에서는 실시간으로 처리하기에 어려움이 존재한다. 또한 kernal size가 클수록 연산 및 데이터양이 많아지는 것을 확인하였다. 제안하는 방식으로 평균을 계산하면 미비한 값 차이지만 연산 횟수를 확연히 줄일 수 있다. 또한 kernal size에 따라 영상 지연이 발생하게 되는데 이는 상황 및 적용하는 제품을

고려하여 kernal size를 정하는 것이 중요하다.

Mean filter는 픽셀 간의 값 변화가 줄어들어 영상의 경계선이 무뎌지고 noise를 줄이는데 주로 사용되는 영상필터이다. 이에 mean filter 기반의 thermal noise filter 적용 시 외부온도에 대한 변화 및 열잡음에 대한 보상이 가능할 것이다. 이는 열센서 기반의 다양한 분야에 활용될 것으로 기대된다.

참고문헌

- [1] Seok-Min Hong, "Advanced LWIR Thermal Imaging Sight Desing," Journal of The Korean of Optics and Photonics, Vol. 16, No. 10, 2005, pp. 209-215.
- [2] R. Olivier, B. Stephan, and B. Pierre, "Non uniformity correction and thermal drift compensation of thermal infrared camera," Proc. SPIE 5405, 2004, pp. 294-302.
- [3] Young-Man Kwon, "Fast Video Stabilization Method Using Integral Image," Journal of The Institute of Internet, Bradcasting and Communication(IIBC), Vol. 10, No. 5, 2010, pp. 13-20.
- [4] Jung-Ho Shin, "Image Correction Method for Uncooled IR TECless Detector with Non-linear characteristic due to Temperature Change," Journal of the Korea Society of Computer and Information, Vol. 22, No. 10, 2017, pp. 19-26.