

# 탐색 영역 데이터의 재사용 및 조기중단이 가능한 HEVC 움직임 추정 엔진 설계

## Design of HEVC Motion Estimation Engine with Search Window Data Reuse and Early Termination

허아름\*, 박태욱\*, 이성수\*  
Ahrum Hur\*, Taewook Park\*, Seongsoo Lee\*

### Abstract

In HEVC variable block size motion estimation, same search window data are duplicatedly used in each block size. It increases memory bandwidth, and it is difficult to exploit early termination. In this paper, largest block size and its corresponding smaller block sizes with same positions are performed at the same time. It reduces memory bandwidth and computation by reusing search window data and computation results. In the early termination, image quality can be degraded when it determines early termination by observing largest block size only, since smaller block sizes cannot be equally terminated due to their relative positions. So, in this paper, processing order of early termination is changed to perform smaller block sizes in turns. The designed motion estimation engine was described in Verilog HDL and it was synthesized and verified in 0.18um process technology. Its gate count and maximum operating frequency are 36,101 gates and 263.15 MHz, respectively.

### 요약

HEVC 가변 블록 크기 움직임 추정 기법에서는 동일 탐색 영역의 데이터가 블록 크기 별로 중복 사용되어 탐색 영역 데이터의 사용량이 많고 조기중단 기법의 적용이 어렵다는 문제점이 있다. 본 논문에서는 최상위 블록 크기 PU를 탐색의 기준으로 두어, 최상위 블록 크기 PU의 탐색과 동일 지점에서의 하위 블록 크기 PU의 탐색을 동시에 진행하여 메모리 대역폭을 줄였고, 최상위 블록 크기 PU의 연산 결과값을 재사용하여 연산량 또한 줄였다. 조기중단 기법을 적용했을 때 또한 최상위 블록 크기 PU를 기준으로 판별하면 하위 블록 크기 PU의 위치에 따라 탐색 기회가 고르게 반영되지 않아 화질의 저하로 이어질 수 있으므로 이를 방지하기 위해 조기중단 기법의 적용 순서를 최상위 블록 크기 PU 내의 최하위 블록 크기 PU를 기준으로 위아래로 번갈아 수행하도록 변형하였다. 설계된 ME 엔진은 Verilog HDL로 기술하고 0.18um 공정에서 합성 및 검증하였으며, 하드웨어 크기는 36,101 게이트이고 최대 동작 속도는 263.15 MHz이다.

*Key words: Motion Estimation, Hardware, Architecture, Variable-Size Block, Data Reuse, Early Termination*

\* School of Electronic Engineering, Soongsil University

★ Corresponding author (e-mail: sslee@ssu.ac.kr, tel: 02-820-0692)

※ Acknowledgment

“This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Korea (2014R1A1A2059625).”

Manuscript received Jun. 22, 2016; accepted Aug. 29, 2016

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

## I. 서론

최근 고화질 멀티미디어 영상에 대한 관심과 수요가 크게 증가함에 따라 제안된 국제 표준인 HEVC(high efficiency video coding)<sup>[1]</sup>는 압축률을 높이면서도 연산량을 줄이기 위해 움직임 추정(ME: motion estimation)에서 가변 블록 크기 움직임 추정, 부화소 움직임 추정, 조기중단 기법 등의 다양한 방법을 이용한다.

영상 압축 표준은 적용 대상에 따라 소프트웨어와 하드웨어의 형태로 구현된다. 이때 고해상도 실시간 처리를 위해서는 높은 연산량을 감당하기 위해 하드웨어로 구현하는 것이 일반적이다. 하지만 HEVC의 움직임 추정 기법<sup>[2][3]</sup>의 경우, 가변 블록 크기 움직임 추정 기법을 사용하여, 각 블록 크기마다 독립적 움직임 추정을 수행한다. 이는 하드웨어로 구현 시, 동일 탐색 영역의 데이터가 블록 크기 별로 중복 사용되어 탐색 영역 데이터의 사용량이 많고 조기중단 기법의 적용이 어렵다는 문제점을 지닌다. 특히 영상 크기가 커지는 경우, 하드웨어 구현에서조차 실시간 수행을 보장하기가 어렵다.

실시간 수행이 가능할 수준으로 연산량과 메모리 대역폭을 줄이기 위해 TZS(test zone search)<sup>[4]</sup>와 같은 다양한 움직임 추정 기법이 제안되었지만 이 또한 각 가변 블록 크기마다 따로 움직임 추정을 수행하기 때문에 메모리 대역폭과 화소 비교 연산량을 만족할만한 수준으로 줄이지 못하고 있다.

이러한 문제점을 해결하기 위해서 가변 블록 크기를 사용하면서도 상위 크기 블록 탐색과 하위 크기 탐색의 위치를 일치시켜서 상위 크기 블록을 탐색하는 동안 얻어지는 중간값을 하위 크기 블록에도 사용하는 기법<sup>[5]</sup>이 제안되었다. 이 기법은 조기중단 기법(ET: early termination)<sup>[6]</sup>도 적용이 가능해서 기존 기법에 비해 메모리 접근량과 연산량을 크게 줄일 수 있다.

본 논문에서는 [5]에 제안된 움직임 추정 기법을 효과적으로 수행할 수 있는 아키텍처를 제안하고 이를 구현하였다. 구현된 움직임 추정 엔진은 하드웨어 크기가 매우 작아 HEVC 실시간 움직임 추정이 가능하며 TZS를 비롯한 다양한 움직임 추정 기법에 쉽게 적용이 가능하다.

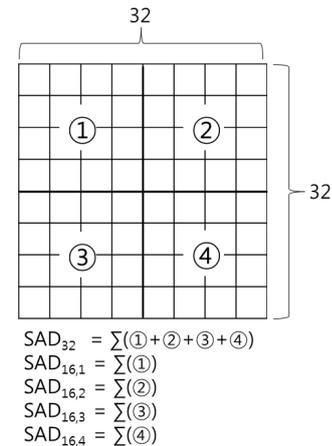


Fig. 1. Example of data reuse in ME

그림 1. ME에서의 데이터 재사용 예

## II. 조기중단 기법을 적용한 움직임 추정 하드웨어 제안

본 논문에서는 그림 1과 같이 최상위 블록 크기 PU(prediction unit)의 탐색을 수행할 때 여기에서 얻어진 중간 계산값을 동일 탐색 지점(search position)의 하위 블록 크기 PU의 탐색에 그대로 사용함으로써 하위 블록 크기 PU를 탐색하는데에는 별도의 연산량이 필요하지 않다. 이 과정에서 탐색 영역의 데이터를 재사용하여 메모리 대역폭을 크게 줄일 수 있으며, 화소 비교 연산인 SAD(sum of absolute differences)와 움직임 추정의 시작점인 MVP(motion vector predictor)도 재사용하여 화소 비교 연산량도 크게 줄일 수 있다.

다만 모든 블록 크기의 PU가 같은 위치에서 탐색을 수행하고 실제로는 이들 탐색이 공통으로 수행되므로, 조기중단 기법을 적용했을 때 최상위 블록 크기 PU가 조기중단에 도달하지 않은 상태에서 하위 블록 크기 PU가 조기중단에 도달한 경우에는 조기중단을 수행해야 할지 말아야 할지 판단하기 어려운 문제가 발생한다.

즉, 최상위 블록 크기 PU를 기준으로 전체 PU를 조기중단시키는 경우에는 조기중단 기법이 드물게 일어나서 연산량 감소 효과가 줄어들고, 하위 블록 크기 PU를 기준으로 전체 PU를 조기중단시키는 경우에는 상위 블록 크기 PU가 함께 중단되는 바람에 전체적으로 탐색이 덜 수행되어

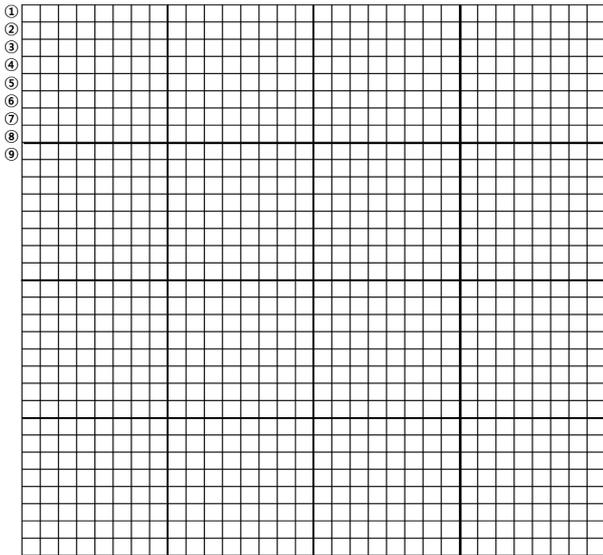


Fig. 2. Conventional row-by-row processing order  
그림 2. 기존 행 단위 연산 순서

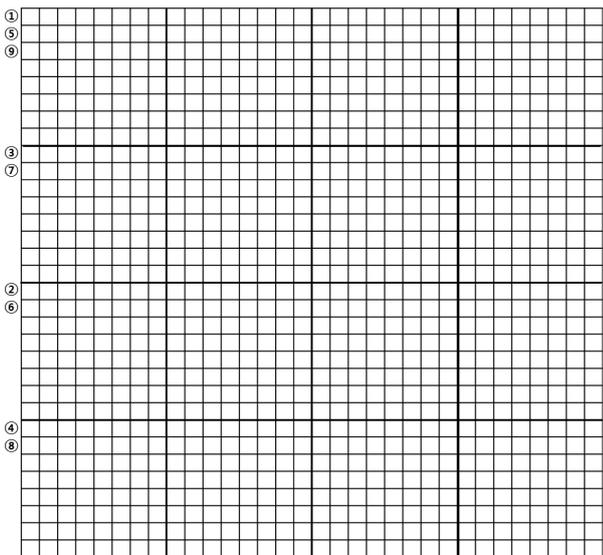


Fig. 3. Change of row-by-row processing order  
그림 3. 연산하는 행의 순서 변경

화질이 나빠질 수 있다. 본 논문에서는 연산량 보다는 화질을 중시하여 최상위 블록 크기 PU가 조기중단될 때 동일 지점의 하위 블록 크기 PU를 함께 중단하도록 하였다.

최상위 블록 크기 PU가 조기중단되면 동일 지점의 하위 블록 크기 PU는 해당 PU의 위치에 따라 일부는 탐색이 완료되고 일부는 최상위 블록 크기 PU의 조기중단에 따라 함께 조기중단된다. 기존의 조기중단 기법은 그림 2와 같이 행 단위로 위에서부터 아래까지 순차적으로 SAD를 계산하면서 그 값이 이전까지 찾은 최소 SAD를

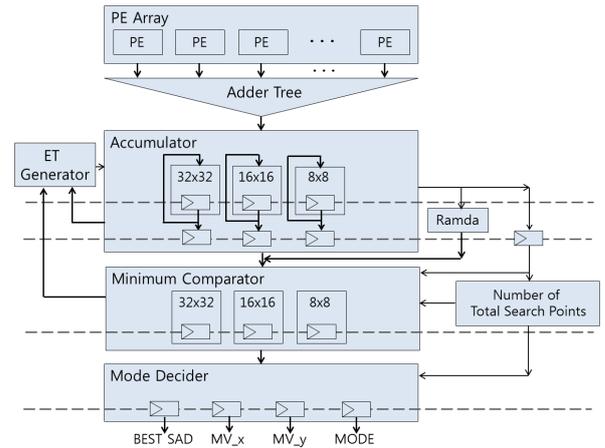


Fig. 4. Overall block diagram of ME engine  
그림 4. ME 엔진 전체 블록도

넘는 경우 조기중단을 수행한다. 이 경우 최상위 블록 크기 PU의 하단부에 위치한 PU들은 최상위 블록 크기 PU에 의해 강제로 함께 조기중단될 확률이 상단부에 위치한 PU에 비해 훨씬 높으며, 하단부에 위치한 PU일수록 조기중단되어 화질이 나빠질 가능성이 높아지게 된다.

이러한 문제점을 보완하기 위해 본 논문에서 제안하는 하드웨어 아키텍처는 최상위 블록 크기 PU에서 수행되는 탐색 순서를 그림 3과 같이 위아래 행의 위치를 번갈아 연산하도록 변형하였다. 이를 통해 모든 최하위 블록 크기 PU가 위치에 상관없이 조기중단 여부 판단에 반영될 수 있어 성능 저하를 줄일 수 있다.

### III. 아키텍처 설계

ME 엔진은 그림 4와 같이 처리기 배열(PE array: processing element array), 덧셈 트리(adder tree), 블록 크기 별 누적기(accumulator), 블록 크기 별 최소값 비교기(minimum comparator), 조기중단 판별기(ET generator: early termination generator) 및 모드 선택기(mode decider)의 6개의 주요 모듈로 구성된다.

처리기 배열은 처리기 16개를 병렬로 사용하여 탐색 지점으로부터 행 방향으로 16개의 화소 위치에서의 참조 화소와 현재 탐색 영역 화소간의 비교값을 한 번에 구한다. 덧셈 트리는 16개의 비교값을 받아 블록 크기 별로 더해 각 블록 크기 별 행의 합으로 만든다. 그렇게 구한 한 행의

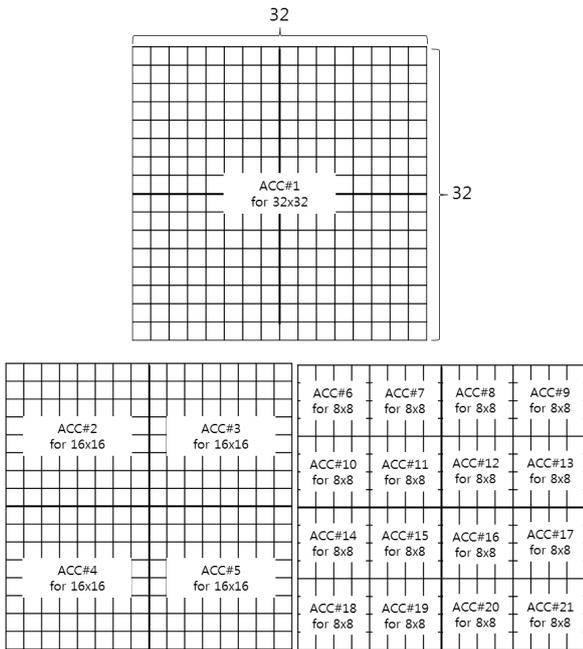


Fig. 5. Assignments of accumulators for variable-size block

그림 5. 각 블록 크기 별 누적기 배치

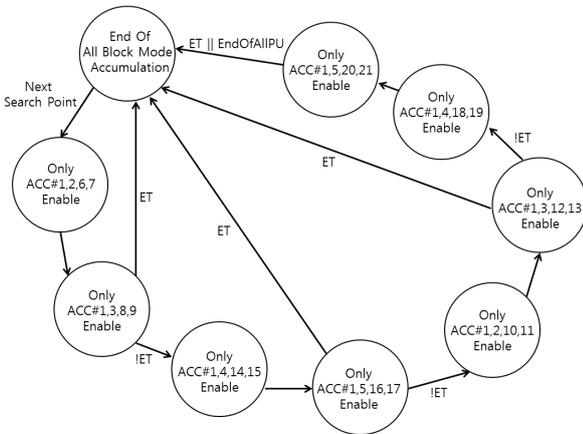


Fig. 6. FSM to control accumulator

그림 6. 누적기 제어 FSM

합을 각 블록 크기 별 누적기로 전달하여 해당 블록 크기에 맞게 누적하고 최종 SAD를 구한다.

그 과정에서 최상위 블록 크기 PU에서 한 행에 대한 총 합이 구해질 때마다 조기중단 판별기에서 이전 탐색 지점에서의 최적값과 비교하며, 매 행마다 조기중단 여부를 판별하여 반영한다. 각 탐색 지점 별 SAD의 최종 값이 구해지면 최소값 비교기에서는 전체 탐색 지점에서의 SAD 최소값을 구하고, 이어지는 모드 선택기에서 전체 PU 및 CU 단위의 최적 SAD 값을 택하여 당

시의 움직임 벡터(MV: motion vector)와 함께 출력한다.

이전 모듈인 덧셈 트리에서 나오는 값은 블록의 한 행밖에 되지 않으므로 블록 당 전체 SAD 합을 구하는 누적기가 필요하다. 본 설계에서 누적기는 그림 5과 같이 각 블록 크기 별로 ACC#1부터 ACC#21로 구성하였다. 32x32 크기의 PU 연산에는 ACC#1을, 16x16 크기의 PU 연산에는 ACC#2 ~ ACC#5를 사용하며 나머지 8x8 크기의 PU 연산은 ACC#7 ~ ACC#21을 이용한다.

각 ACC를 하나로 묶는 상위 모듈에서 FSM을 이용해 그림 6과 같은 순서로 각 ACC를 동작하게 한다. 이때 조기중단 여부는 최상위 블록 크기 PU인 32x32 블록을 기준으로 적용한다. 즉, 16개의 PE에서 16개의 화소 연산값의 합이 두 번 나와 32x32 크기의 블록 한 행의 합이 구해질 때마다 조기중단 조건을 검사하며 그 판별 여부에 따라 ET 신호로 전체 동작을 제어한다.

ET 신호 생성 시, 그림 6에서 ACC#1,2,6,7과 ACC#1,3,8,9를 차례로 동작하도록 제어한 후에 ACC#1,4,14,15와 ACC#1,5,16,17을 차례로 동작하도록 하는 것을 확인할 수 있는데, 해당하는 ACC의 위치를 그림 5에서 확인해보면 최하위 블록 크기 PU인 8x8 블록이 고르게 연산에 반영될 수 있도록 최상위 블록 크기 PU인 32x32 블록에서 위아래로 연산 순서를 바꾸어 수행함을 알 수 있다.

#### IV. 시뮬레이션 결과

제안한 ME 엔진의 성능을 분석하기 위한 시뮬레이션은 HEVC 소프트웨어 모델인 HM 13.0<sup>[7]</sup>을 기반으로 수행하였다. CTU(coding tree unit)의 크기는 32x32로 설정하였으며 움직임 추정 알고리즘은 TZS를 사용하였다. PU는 32x32, 16x16 및 8x8의 세 가지 블록 크기 모드를 사용하였고, 가장 큰 블록 크기인 32x32의 배수로 이루어진 크기의 테스트 영상인 A 클래스(2560x1600 화소, 30 프레임/초, 150 프레임, 4 Mbps 전송 비트율)의 Traffic, PeopleOnStreet 영상과 C 클래스(832x480 화소, 30 프레임/초, 300 프레임, 512 Kbps 전송 비트율)의 Keiba, RaceHorses, Mobisode2 영상을 사용하였다.

Table1. Simulation results of class A

표 1. A 클래스 시뮬레이션 결과

Algorithm			Sequence	
			Traffic	PeopleOnStreet
Original TZS	ET Disable	Required Frequency (MHz)	627.79	1072.79
		PSNR(dB)	37.0493	29.0129
Proposed TZS	ET Disable	Required Frequency (MHz)	222.90	430.60
		PSNR(dB)	36.9771	28.7982
	ET Enable	Required Frequency (MHz)	150.62	311.80
		PSNR(dB)	36.9621	28.7883

Table2. Simulation results of class C

표 2. C 클래스 시뮬레이션 결과

Algorithm			Sequence		
			Keiba	Race Horses	Mobisode2
Original TZS	ET Disable	Required Frequency (MHz)	115.75	128.26	124.36
		PSNR(dB)	33.9672	29.6632	43.4396
Proposed TZS	ET Disable	Required Frequency (MHz)	46.74	56.01	53.32
		PSNR(dB)	33.8167	29.4976	43.4292
	ET Enable	Required Frequency (MHz)	34.53	42.11	41.04
		PSNR(dB)	33.8081	29.4766	43.4090

시뮬레이션 결과는 표 1 및 표 2와 같다. 제안한 ME 엔진은 가변 블록 크기 움직임 추정 시 최상위 블록 크기 PU의 탐색을 수행할 때 다른 블록 크기 PU도 함께 탐색되며, 변형된 조기중단 기법까지 적용하면 필요 연산량이 크게 줄어든다. 따라서 움직임 추정을 실시간으로 완료하기 위해 필요한 ME 엔진의 동작 주파수를 기존에 비해 1/4.1 ~ 1/3.4로 크게 줄일 수 있다. 이때 PSNR(peak signal-to-noise ratio) 저하는 0.03 ~ 0.22dB로 무시할만한 수준임을 확인할 수 있다.

제안한 ME 엔진을 Verilog HDL로 설계하고

ModelSim을 이용하여 검증하였다. 검증에 필요한 모든 계수 및 데이터 값은 HM13.0 소프트웨어에서 8 bits 이진수로 추출하여 사용하였으며, 최종 출력된 결과 값이 HM13.0 기반 시뮬레이션의 결과와 일치함을 확인할 수 있었다. 구현된 ME 엔진은 검증 시 데이터를 입력 받는 부분을 별도 모듈로 구현하여 그 형태와 구조를 자유롭게 변형할 수 있어 추후 다른 움직임 추정 기법 알고리즘의 적용에도 용이하다. 이를 0.18um 공정에서 Design Compiler를 사용하여 합성 시, 약 36,101 게이트로 나타나며 최대 동작 속도는 263.15 MHz이다. 구현된 ME 엔진이 2K 영상인 A 클래스 영상을 처리하기 위해 필요한 동작 속도가 150 ~ 312 MHz 정도이기 때문에 0.13um나 그 이상의 공정을 사용하면 2K급 영상까지 무리 없이 동작할 수 있을 것으로 보인다.

## V. 결론

본 논문에서는 조기중단 기법의 적용이 가능한 가변 블록 크기 움직임 추정 엔진을 구현하였다. 그 과정에서 각 블록 크기별 움직임 추정을 동시에 수행하여 메모리 대역폭을 줄였고, 최상위 블록 크기 PU에서의 SAD 및 PMV를 재사용 하여 연산량 또한 줄였다. 여기에 조기중단 기법의 수행 순서를 위아래로 바꾸어 적용하여 최하위 블록 크기 PU의 탐색 기회를 보장하였다.

구현된 하드웨어는 약 36,101 게이트로 구성되며 최대 263.15 MHz 속도의 동작이 가능하다. HEVC 표준 영상을 시뮬레이션 할 경우 조금만 더 좋은 공정을 사용하면 2560x1600 크기의 2K 영상까지 동작이 가능하다.

## References

- [1] G. Sullivan, J. Ohm, W. Han, T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard", *IEEE Trans. Circuits Syst. Video Technol.* vol. 22, no. 15, pp. 1649-1668, Sep. 2012.
- [2] H. Yang and S. Lee, "Motion Estimation Algorithm to Guarantee Hard Realtime

Operation”, *j.inst.Korean.electr.electron.eng*, vol. 17. no. 1, pp. 36-43, Mar. 2013.

[3] H. Bae and J. Chong, “A Simple Block-Based Motion Estimation Algorithm for Discontinuity Blocks”, *j.inst.Korean.electr.electron.eng*, vol. 6. no. 1, pp. 94-101, Jun. 2002.

[4] JVT of ISO/IEC MPEG and ITU-T VCEG, “MVC Software Reference Manual - JMVC 8.2”, <http://sourceforge.net/projects/tortoiseCVS/files/tortoiseCVS-stable/TortoiseCVS-1.12.5.exe/download>

[5] T. Park, A. Hur, and S. Lee, “Reusing Search Window Data and Exploiting Early Termination in Variable Block Size Motion Estimation”, *j.inst.Korean.electr.electron.eng*, vol. 20. no. 1, pp. 111-114, Mar. 2016.

[6] D. Guevorkian, A. Launianen, P. Liuha, and V. Lappalainen, “Architecture for the sum of absolute differences operation”, in *Proc. of IEEE Workshop on Signal Processing Systems (SIPS2002)*, 2002, pp. 57-62.

[7] JVT of ISO/IEC MPEG and ITU-T VCEG, “HM Reference Software”, [https://hevc.hhi.fraunhofer.de/svn/svn\\_HEVCSoftware](https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware)

## BIOGRAPHY

### Ahrum Hur (Student Member)



2015 : BS degree in Electronic Engineering, Soongsil University.  
2015~Now : MS candidate in Electronic Engineering, Soongsil University.

### Taewook Park (Student Member)



2014 : BS degree in Electronic Engineering, Soongsil University.  
2016 : MS degree in Electronic Engineering, Soongsil University.

### Seongsoo Lee (Life Member)



1991: BS degree in Electronic Engineering, Seoul National University.  
1993: MS degree in Electronic Engineering, Seoul National University.

1998: PhD degree in Electrical Engineering, Seoul National University.

1998~2000: Research Associate, University of Tokyo.

2000~2002: Research Professor, Ewha Womans University.

2002~Now: Professor in School of Electronic Engineering, Soongsil University.