JKIICE Journal of the Korea Institute of Information and Communication Engineering

한국정보통신학회논문지(J. Korea Inst. Inf. Commun. Eng.) Vol. 21, No. 3 : 594~600 Mar. 2017

고성능 HEVC 부호기를 위한 움직임추정 하드웨어 설계

박승용 · 전성훈 · 류광기*

The Design of Motion Estimation Hardware for High-Performance HEVC Encoder

Seungyong Park · Sunghun Jeon · Kwangki Ryoo*

Department of Information and Communication Engineering, Hanbat National University, Daejeon 34158, Korea

요 약

본 논문에서는 고성능 HEVC(High Efficiency Video Coding) 부호기를 위한 전역탐색 기반의 움직임추정 알고리즘과 이에 적합한 하드웨어 구조를 제안한다. HEVC 화면 간 예측에서의 움직임추정은 시간적 중복성을 제거하기 위하여 보간 된 참조 픽쳐에서 현재 PU와 상관도가 높은 예측 블록을 탐색하는 과정으로 전역탐색 알고리즘과 고속탐색 알고리즘을 이용한다. 전역 탐색 기법은 주어진 탐색 영역내의 모든 후보 블록에 대하여 움직임을 예측하기 때문에 최적의 결과를 보장하지만 연산량 및 연산시간이 많은 단점을 지닌다. 그러므로 본 논문에서는 Inter Prediction의 연산량 및 연산시간을 줄이기 위해 전역탐색에서 SAD연산을 재사용하여 연산복잡도를 줄이는 새로운 알고리즘을 제안하고 이에 적합한 하드웨어 구조를 제안한다. 제안된 알고리즘은 HEVC 표준 소프트웨어 HM16.12에 적용하여 검증한 결과 기존 전역탐색 알고리즘보다 연산시간은 61%, BDBitrate는 11.81% 감소하였고, BDPSNR은 약 0.5% 증가하였다. 또한 하드웨어설계 결과 최대 동작주파수는 255 Mhz, 총 게이트 수는 65.1K 이다.

ABSTRACT

This paper proposes a global search based motion estimation algorithm for high performance HEVC encoder and its hardware architecture. To eliminate temporal redundancy, motion estimation in HEVC inter-view prediction uses global search and fast search algorithm to search for a predicted block having a high correlation with the current PU in an interpolated reference picture. The global search method predicts the motion of all candidate blocks in a given search area, thus ensuring optimal results, but has a disadvantage of large computation time. Therefore we propose a new algorithm that reduces computational complexity by reusing SAD operation in global search to reduce computation time of inter prediction. As a result of applying the proposed algorithm to standard software HM16.12, the computation time was reduced by 61%, BDBitrate by 11.81%, and BDPSNR by about 0.5% compared with the existing search algorithm. As a result of hardware design, the maximum operating frequency is 255 MHz and the total number of gates is 65.1K.

키워드 : HEVC, 움직임추정, Encoder, 전역탐색

Key word : HEVC, Motion Estimation, Encoder, Full Search

Received 09 January 2017, **Revised** 16 January 2017, **Accepted** 26 January 2017

* **Corresponding Author** Kwangki Ryoo(E-mail:kkryoo@gmail.com, Tel:+82-42-821-1710)
Department of Information and Computer Engineering, Hanbat National University, Daejeon 34158, Korea

Open Access http://doi.org/10.6109/jkiice.2017.21.3.594 print ISSN: 2234-4772 online ISSN: 2288-4165

ⓒThis is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(http://creativecommons.org/li-censes/ by-nc/3.0/) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright Ⓒ The Korea Institute of Information and Communication Engineering.

Ⅰ. 서 론

HEVC는 H.264/AVC의 표준화를 수행하였던 ISO/IEC의 MPEG(Moving Picture Expert Group)과 ITU-T의 VCEG(Video Coding Expert Group)이 공동으로 개발한 차세대 동영상 압축 표준 기술이다[1]. HEVC에서는 CU(Coding Unit), PU(Prediction Unit), TU(Transform Unit) 3가지 종류의 부호화 단위 적용과 계층적 쿼드-트리(Quad-tree) 구조의 부호화 수행을 하며, 64×64부터 8×8까지 다양한 크기의 부호화 단위를 사용한다. 이외에도 화면 내 예측 방향의 증가, 향상된 움직임 예측 기법, 움직임 벡터 병합 및 세분화된 루프 필터를 적용하여 이전의 압축코덱인 H.264/AVC 대비 약 2배의 압축 성능을 보이지만 다양한 부호화 구조와 향상된 예측 기법에 따른 연산 복잡도가 크게 증가하였다.

새로운 기술들 중 움직임예측은 현재 블록과 가장 유사한 예측 블록을 생성한다. 현재 PU와 참조 블록의 상관도를 비교하는 과정에서 정소화소와 부화소의 특징을 고려하여 간략화 된 상관도 측정 방법인 SAD(Sum of Absolute)를 사용하지만 화면 간 예측의 전역탐색 알고리즘의 경우 4×8 PU부터 최대 64×64 PU크기까지 다양한 크기의 PU에 대한 SAD 연산을 반복 수행하기 때문에 연산량 및 연산 시간이 많다. 따라서 본 논문에서는 고성능 HEVC 부호기를 위해 반복 연산되는 SAD 연산의 연산량 및 연산 시간을 줄이는 새로운 알고리즘을 제안하고 적합한 하드웨어 구조를 제안한다. 제안하는 HEVC 움직임 추정 알고리즘 및 하드웨어 구조는 연산 시간 감소와 처리율 향상을 위해 전역탐색 과정에서 4×4 단위의 SAD연산 결과를 저장하여 재사용하도록 설계하였다. 또한 최적의 PU분할 블록선택을 병렬적으로 처리하도록 설계함으로써 하드웨어 면적 및 연산시간을 효과적으로 감소시켰다.

Ⅱ. HEVC 화면 간 예측

2.1. 움직임추정

화면 간 예측의 움직임추정(Motion Estimation ; ME)은 인코더에서 수행되는 과정으로 참조픽쳐 에서 현재 PU와 상관도가 높은 예측 블록을 탐색하는 과정이다. 움직임 추정 수행 결과 PU 단위로 참조픽쳐 리스트의 정보, 참조 픽쳐 인덱스, 움직임 벡터와 차분신호를 변환 양자화 한 계수가 디코더로 전송된다. 디코더에서는 인코더로부터 전송된 주변 정보를 이용하여, 인코더와 동일한 예측 블록을 생성하고 양자화 된 잔차신호를 사용하여 복원 블록을 생성하는 움직임 보상 과정을 수행한다[2]. 그림 1은 움직임 추정 과정을 나타낸다.

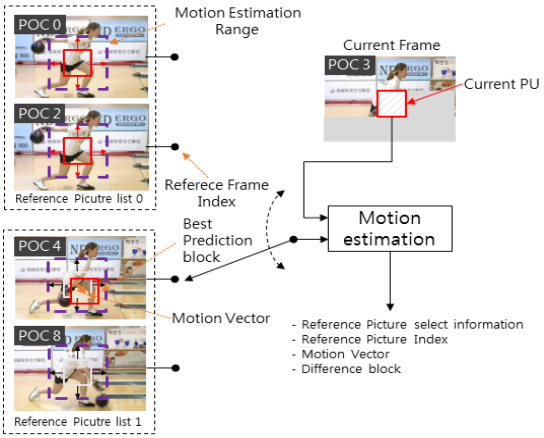

Fig. 1 HEVC Motion Estimation process

현재 PU와 참조 블록의 상관도를 비교하는 과정에서 간략화 된 상관도 측정 방법인 SAD(Sum Absolute Different) 수식(1)을 사용한다.

$$SAD(i,j,k,l) = \sum |B_{cur}(i,j) - B_{ref}(k,l)| \qquad (1)$$

B_{cur}는 현재 블록, B_{ref}는 참조 픽쳐 내에 존재하는 움직임 추정 후보 블록, i, j는 현재 PU의 위치 k, l은 움직임 추정 대상의 PU 위치를 나타낸다. 정수 화소의 움직임 추정은 현재 블록과 참조 블록의 차분값의 절대 값을 최소화하는 참조 블록을 선택함으로써 하나의 예측 블록을 선택한다.

2.2. 움직임추정 알고리즘

HEVC에서는 최적의 예측블록을 찾고자 할 때, Test Zone Search(TZS) 알고리즘과 Full-Search 알고리즘을 사용한다. Full-search 의 경우 그림 2와 같이 X, Y 픽셀 좌표 (- 64, -64) ~ (64, 64)까지 탐색을 반복 수행하기 때문에 탐색 성능은 뛰어나지만 탐색범위가 커지면 연산량이 매우 커지는 단점이 있다. 반면 TZS알고리즘은

grid search로 단계별 4개 혹은 8개씩 Search point 들만 계산하기 때문에 연산속도는 빠르지만 Full-Search에 비해 예측 성능이 저하되는 단점이 있다[3].

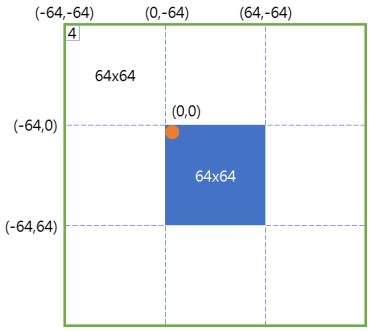

Fig. 2 64×64 PU block full-search Search range

Ⅲ. 제안하는 움직임 추정 알고리즘

본 논문에서는 Full-Search 알고리즘에서 연산량을 효과적으로 줄이기 위해 4×4 단위의 SAD 연산 결과를 재사용하여 전체 PU블록에 대한 SAD 결과를 계산하는 알고리즘을 제안한다. 그림 3과 같이 4개의 4×4 블록의 SAD 결과 값의 합은 8×8블록의 SAD 결과 값과 같으며, 8×8 블록 4개의값은 1개의 16×16 블록의 SAD 결과 값과 같다. 따라서 4x4 단위 SAD 결과 값을 저장하고 필요한 영역(PU분할 크기) 만큼의 값만 호출하여 값을 더해줌으로써 반복 되는 SAD 연산횟수를 효과적으로 줄이고 모든 PU분할에 대해 SAD 값을 구할 수 있다.

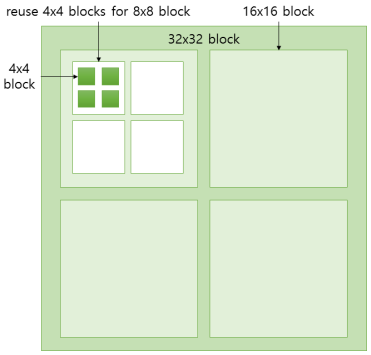

Fig. 3 Block reuse method

그림 4는 블록 재사용을 위해 제안하는 화면 간 예측 전체 흐름을 나타낸다. xCheckRDCostMerge2N×2N에서 skip, Merge 에 대한 RD-Cost를 계산한다. 이후 2N×2N, N×N, N×2N, 2N×N등의 PU mode 순서대로 RD-Cost를 계산하는 대신 2N×2N SAD 연산을 수행할 때 4×4단위로 SAD 결과를 메모리에 저장한 뒤 차례로 N×N, N×2N, 2N×N PU mode 순서대로 해당 영역만큼의 memory index를 호출하여 결과 값을 계산한다. 마지막으로 AMP(Asymmetric Motion Partion) 조건에 따라 비대칭적인 Partition 에 대해 RD cost 계산하는 과정에서도 해당 영역만큼의 memory index를 호출하여 최적의 PU mode를 결정한다.

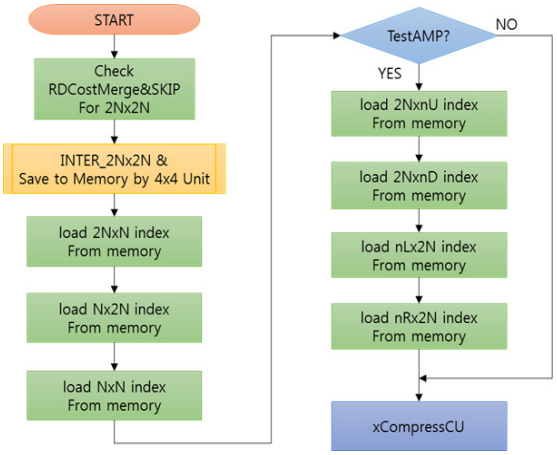

Fig. 4 proposed algorithm

본 논문에서 제안하는 SAD 알고리즘의 성능 향상을 비교하기 위해서 HEVC 참조 소프트웨어인 HM-16.12을 이용하여 모든 클래스 영상에 대해 검증하였다. 성능 평가지표는 4가지 QP(22, 27, 32, 37)에 대한 BD-PSNR과 BD-Bitrate를 사용하였다. 표 1은 실험결과를 나타낸다.

[4], [5]의 성능 비교를 진행한 결과 제안하는 SAD 연산 알고리즘은 표준 소프트웨어 대비 평균적으로 61% 인코딩 속도가 향상되었으며 수식 (2)와 같이 계산하였다.

$$\triangle TS(\%) = \left(\frac{TS_{HM} - TS_{propose}}{TS_{HM}} \right) \times 100 \qquad (2)$$

Table. 1 Comparison of HM-16.12 standard software and proposed algorithm results

Sizes	Sequences	proposed			[4]		[5]	
		BDPSNR	BDBitrate	△TS(%)	BDBitrate	△TS(%)	BDBitrate	△TS(%)
Class A	Traffic	0.49	-12.26	60.5	1.88	60.2	1.67	45.3
	PeopleOnstreet	0.18	-3.76	59.3	1.11	41.2	1.44	29.2
Class B	Kimono	1.36	-30.02	60.4	1.93	53.2	1.04	36.7
	ParkScene	0.40	-11.34	62.3	2.09	55.2	0.86	44.4
	Cactus	0.31	-11.21	62.3	1.65	55.5	1.20	39.7
	BasketballDrive	1.51	-44.55	62.2	2.22	56.6	1.27	40.4
Class C	RaceHorsesC	1.16	-21.18	55.4	1.29	40.9	1.13	26.8
	BQMall	0.23	-4.66	52.8	1.70	52.6	1.27	42.2
	PartyScene	0.02	-0.26	57.3	2.47	40.3	0.99	31.2
	BasketballDrill	0.27	-5.98	54.6	0.96	48.3	1.58	37.2
Class D	RaceHorses	1.17	-18.70	70.2	1.46	30.3	0.97	23.0
	BlowingBubbles	0.01	-0.21	71.2	2.82	42.6	1.17	32.7
	BasketballPass	0.06	-1.14	60.0	3.30	53.1	0.84	40.7
	BQSquare	0.04	0.00	70.0	0.75	52.0	1.30	32.6
Average	-	0.515	-11.81	61.32	1.83	48.7	1.20	35.9

Ⅳ. 제안하는 하드웨어 구조

32×32 화소 단위로 SAD연산을 수행할 때 발생하는 높은 연산량과 연산시간을 감소시키고 하드웨어 면적을 최소화하기 위해, 4x4 연산기의 병렬적 구조를 제안하였다. 기존 움직임 추정 SAD연산기의 구조는 top-down 방식으로 최적의 SAD값을 계산하여 PU블록 분할을 결정하기 위해 LCU(Largest CU) 블록을 표 2와 같이 최대 3-depth까지 하위 블록으로 분할하여 반복 수행한다[6].

Table. 2 Inter Prediction PU partition

PU SIZE	Depth		
	0	1	2
2Nx2N	32x32	16x16	8x8
2NxN	32x16	16x8	8x4
Nx2N	16x32	8x16	4x8
NxN	16x16	8x8	-
2NxnU	32x8	16x4	-
2NxnD	32x24	16x12	-
nLx2N	8x32	4x16	-
nRx2N	24x32	12x16	-

본 논문에서 제안하는 움직임 추정 구조는 bottom-up 방식으로 최하위 depth를 먼저 연산하여 상위 depth까지 이전에 계산된 연산결과를 재사용하는 구조로 설계하였다.

최하위 depth인 4×4부터 SAD 연산을 수행하여, 4×4의 SAD 결과들을 8×8에서 사용하는 bottom-up 구조를 적용한 결과 top-down 방식일 때는 32x32 블록을 처리를 해야 하므로 1024개의 픽셀을 처리하는 반면, bottom-up 방식은 4x4 블록을 병렬적으로 처리하기 때문에 16개의 픽셀만 처리하면 된다. 따라서 하드웨어 크기와 연산량 및 연산 시간이 크게 줄어든다.

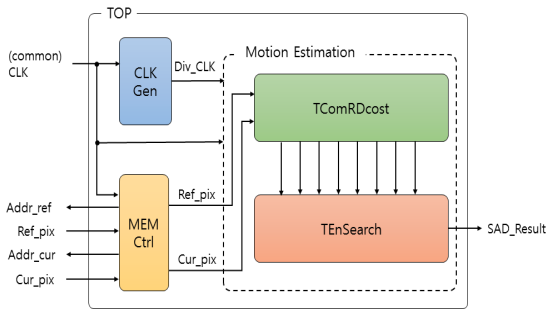

Fig. 5 Proposed motion estimation hardware architecture

그림 5는 제안하는 움직임 추정 하드웨어 구조의 전체적인 Block diagram을 나타낸다. 전체 구조는 필요한 클록을 분주하는 CLKGen 모듈, 메모리로부터 화소들을 입력받기 위한 MemCtrl 모듈, 입력받은 픽셀정보를 4×4 블록단위로 SAD를 계산하여 값을 저장하는 TComRDCost 모듈, 4×4 블록단위로 SAD값이 계산된 정보를 이용하여 모든 PU블록 분할에 대하여 SAD 연산결과를 구하는 TEnSearch 모듈로 구성된다.

4.1. TComRDCost 블록

그림 6은 TComRDCost 블록의 내부 구조를 나타낸다. 제안하는 움직임 추정 하드웨어 구조의 TComRDCost 모듈은 4x4 블록 단위로 SAD 연산을 수행하며 LCU크기 32×32를 4×4 단위로 병렬 처리하기 위해서 64개의 연산기로 구성된다. 각 SAD Computing 모듈에서는 Current block 4×4픽셀, Reference block 4×4 픽셀 32개 값에 대하여 각각 abs(Absolute) 연산을 수행한 뒤 SAD 결과 값을 출력한다.

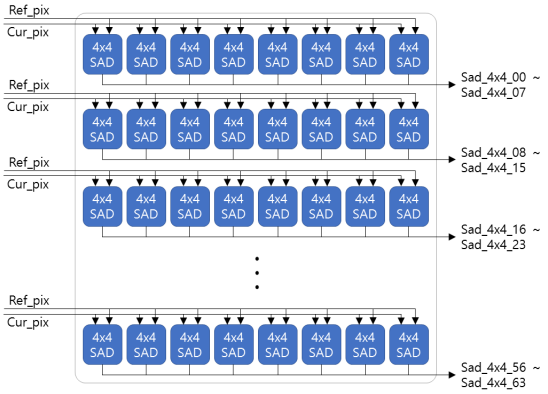

Fig. 6 TComRDCost Block

4.2. TEnSearch 블록

그림7은 TEnSearch 블록의 내부 구조를 나타낸다. TEnSearch블록에서는 TComRDCost로부터 입력 받은 4×4 단위 SAD 값을 이용하여 bottom-up 방식으로 32×32 블록에 대한 모든 분할 PU에 대하여 SAD 결과를 계산한다. 입력받은 64개의 4×4 SAD 연산결과를 이용하여 각 Depth에 따른 SAD 결과를 계산한다. Depth_3 블록에서 8×4, 4×8, 8×8 블록에 대한 SAD 값을 계산한다. 이후 Depth_3의 값은 Depth_2에서 Depth_2의 값은 Depth_1의 값을 계산하는데 재사용되어 bottom-up 계산을 한다. 움직임 추정 알고리즘에서 최적의 PU 블록분할을 결정하고 움직임 벡터를 추정할 때, 최소값의 SAD 결과를 갖는 PU블록을 결정하여 움직임을 추정하기 때문에 Save_min블록에서 각 블록에 대하여 최소 SAD연산결과를 갖는 값만 SAD_Result로 출력된다.

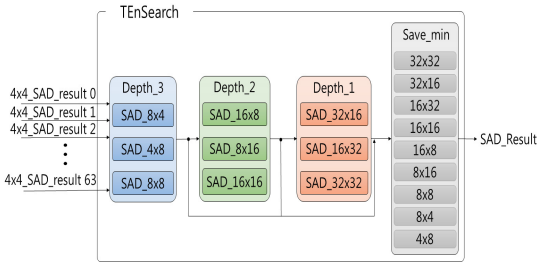

Fig. 7 TEnSearch Block

Ⅴ. 하드웨어 검증 및 결과

제안하는 하드웨어 구조의 검증 방법은 그림 8과 같이 HEVC 참조 소프트웨어인 HM 16.12 에서 32x32 블록의 픽셀값을 추출하여 HM 16.12와 제안하는 하드웨어 입력을 통해 결과값을 비교하였다.

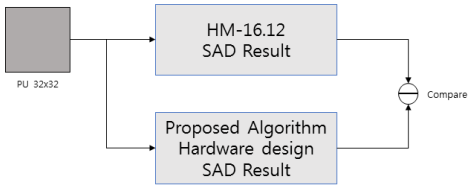

Fig. 8 Comparison between HM 16.12 and proposed hardware

제안하는 하드웨어 구조는 65nm CMOS 공정 라이브러리를 이용하여 IDEC으로부터 지원받은 Synopsys 사의 Design Compile 툴을 통해 합성한 결과 65.1k의 로직 게이트로 구현되며 최대 255 MHz에서 동작함을 검증하였다. 기존에 제안된 [7] 구조는 4×4 블록에서 16×16 블록까지 지원하는 SAD 연산기 구조를 가진다. 또한, 파이프라인 구조를 이용하여 한 사이클당 1×16 픽셀을 입력받으며, 총 16 사이클에 동안 16×16 블록

에 해당하는 픽셀 데이터를 입력받아 처리하는 구조이다. [8]의 구조는 4×4 블록에서 64×64 블록까지 병렬 SAD 유닛을 사용하는 구조 이다. 또한, Xilinx Virtex5 FPGA를 사용하여 합성하였다.

표 3에서 기존에 제안되었던 구조와 비교한 결과, 제안하는 구조의 설계가 게이트 수는 기존에 [7] 구조에 비하여 15% 감소하였으며, 지원 모드는 32×32 블록까지 지원한다. [8]의 구조에 비해 제안하는 구조의 게이트 수는 7배 감소하였지만 [8]의 구조는 64×64 블록까지 지원한다.

Table. 3 Comparison of existing and proposed hardware

Architecture		[7]	[8]	proposed
Technology (FPGA : Virtex5)		180 nm CMOS	65 nm FPGA	65 nm CMOS
Max. frequency		232 MHz	172 MHz	255 MHz
Supporting mode		4×4 to 16×16	4×4 to 64×64	4×4 to 32×32
Gate Counts	CMOS	84.1k	-	65.1k
	FPGA	-	9.18k	1.28k

Ⅵ. 결 론

본 논문에서는 HEVC 움직임 추정 알고리즘에서 수행되는 SAD 연산시간과, 연산량을 최소화하기 위해 전역탐색 기반의 새로운 알고리즘과 이에 적합한 하드웨어 구조를 기술하였다. 제안하는 움직임 추정 알고리즘은 기존 Full-Search 알고리즘에서 발생하는 SAD연산 복잡도와 반복을 줄이기 위해 4×4 단위로 계산된 SAD 결과를 재사용하여 모든 블록에 재사용 한다. 또한, 4×4 단위의 픽셀을 병렬적으로 처리함으로써 효과적으로 하드웨어 면적 및 처리속도를 향상시켰다. 제안하는 알고리즘을 HM표준 소프트웨어 HM-16.12에 적용된 Full-Search 알고리즘과 비교한 결과 기존대비 평균적으로 61% 인코딩 속도가 향상되었으며, BDPSNR은 0.5 증가, BDBitrate는 11.81 감소하였다. 제안하는 하드웨어 구조의 합성 결과는 65.1k의 로직 게이트와 최대 255 MHz 동작 주파수를 가진다.

ACKNOWLEDGMENTS

This research was supported by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the Global IT Talent support program(IITP-2016-R0134-16-1019) and Human Resource Development Project for Brain scouting program(IITP-2017-R2418-17-0007) supervised by the IITP(Institute for Information and Communication Technology Promotion.

REFERENCES

[1] H. Jung, and K. Ryoo, “An Intra Prediction Hardware Architecture Design for Computational Complexity Reduction of HEVC Decoder,” *Journal of the Korea Institute of Information and Communication Engineering*, vol. 17, no. 5, pp. 1203-1212, May 2013.

[2] S. Park, H. Jo, and K. Ryoo, “The Design of Transform and Quantization Hardware for High-Performance HEVC Encoder,” *Journal of the Korea Institute of Information and Communication Engineering*, vol. 20, no. 2, pp.327-334, Feb. 2016.

[3] N. Purnachand, L. N. Alves, and A. Navarro, “Improvements to TZ search motion estimation algorithm for multiview video coding,” in *International Conference on System Signals and Image Processing(IWSSIP) 19th*, Apr. 2012.

[4] J. Zhang, B. Li, and H. Li, “An Efficeient Fast Mode Decision Method for Inter Prediction in HEVC,” *IEEE Transactions Circuits System Video Technology*, vol. 26, no. 8, pp. 1502-1515, Aug. 2015.

[5] S. Ahn, M. Kim, and S. Park. “Fast decision of CU partitioning based on SAO parameter, motion and PU/TU split information for HEVC,” in *Proceeding of Picture Coding Symposium*, pp. 113-116. Dec. 2013.

[6] F. Sampaio, S. Bampi, M. Grellert, L. Agostini, and J. Mattos, “Motion vectors merging: low complexity prediction unit decision heuristic for the inter-prediction of HEVC encoders,” in *IEEE International Conference on Multimedia and Expo* (ICME), pp. 657-662, July. 2012.

[7] Z. Liu, S. Goto, and T. Ikenaga, “Optimization of Propagate Partial SAD and SAD tree motion estimation hardwired engine for H.264,” *IEEE International Conference on Computer Design*, pp. 328-333, Oct. 2008.

[8] P. Nalluri, L. N. Alves, and A. Navarro, “High Speed SAD Architectures for Variable block size Motion Estimation In HEVC Video Coding,” in *Proceeding. IEEE International Conference on ICIP* Circuits, pp. 1233- 1237, Oct. 2014.

박승용(Seungyong Park)

2010년 한밭대학교 공과대학 정보통신공학과 공학사
2012년 한밭대학교 정보통신전문대학원 정보통신공학과 공학석사
2012년 ~ 현재 한밭대학교 정보통신전문대학원 정보통신공학과 박사과정
※ 관심분야 : SoC 플랫폼 설계 및 검증, 멀티미디어 코덱 설계

전성훈(Sunghun Jeon)

2015년 한밭대학교 공과대학 정보통신공학과 공학사
2017년 한밭대학교 정보통신전문대학원 정보통신공학과 공학석사
※ 관심분야 : SoC 플랫폼 설계 및 검증, 멀티미디어 코덱 설계

류광기(Kwangki Ryoo)

1986년 한양대학교 공과대학 전자공학과 공학사
1988년 한양대학교 대학원 전자공학과 공학석사
2000년 한양대학교 대학원 전자공학과 공학박사
1991년 ~ 1994년 육군사관학교 교수부 전자공학과 전임강사
2000년 ~ 2002년 ETRI 시스템IC설계팀 선임연구원
2010년 ~ 2011년 Univ of Texas at Dallas 방문교수
2003년 ~ 현재 한밭대학교 공과대학 정보통신공학과 교수
※ 관심분야 : 공학교육, SoC 플랫폼 설계 및 검증, 멀티미디어 코덱 설계