

일반논문 (Regular Paper)

방송공학회논문지 제23권 제1호, 2018년 1월 (JBE Vol. 23, No. 1, January 2018)

<https://doi.org/10.5909/JBE.2018.23.1.115>

ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

회전 포레스트 분류기법을 이용한 HEVC 스크린 콘텐츠 화면 내 부호화 조기분할 결정 방법

허 정 환^{a)}, 정 제 창^{a)†}

Fast Partition Decision Using Rotation Forest for Intra-Frame Coding in HEVC Screen Content Coding Extension

Jeonghwan Heo^{a)} and Jechang Jeong^{a)†}

요 약

본 논문에서는 머신러닝을 기반으로 한 조기분할 결정 방법을 통하여 High Efficiency Video Coding (HEVC) Screen Content Coding (SCC) 부호화 기기의 속도를 향상시키는 방법을 제안한다. 현재 HEVC에서는 최적의 부호화 효율을 내기 위해 쿼드트리 블록 분할 과정을 수행한다. 이 과정은 부호화기의 높은 계산 복잡도를 요구하기 때문에 블록 구조를 조기에 결정하여 부호화 속도를 향상시키는 방법으로 고속화 연구가 이루어져 왔다. 하지만 스크린 콘텐츠 부호화는 기존의 자연영상에 맞춰진 부호화 과정과 다른 블록 분할 특성을 보이기 때문에 기존의 조기분할 결정 연구를 적용하기 어렵다. 제안하는 방법은 먼저 스크린 콘텐츠 블록을 분류해 낸 다음 다시 블록분할을 결정하는 방법으로 문제를 해결하였고 SCC 공통 실험 조건에서 3.11%의 BD-BR 증가와 평균 42%의 부호화 시간 감소를 보였다.

Abstract

This paper presents a fast partition decision framework for High Efficiency Video Coding (HEVC) Screen Content Coding (SCC) based on machine learning. Currently, the HEVC performs quad-tree block partitioning process to achieve optimal coding efficiency. Since this process requires a high computational complexity of the encoding device, the fast encoding process has been studied as determining the block structure early. However, in the case of the screen content video coding, it is difficult to apply the conventional early partition decision method because it shows different partition characteristics from natural content. The proposed method solves the problem by classifying the screen content blocks after partition decision, and it shows an increase of 3.11% BD-BR and 42% time reduction compared to the SCC common test condition.

Keyword : HEVC, Screen Content Coding, Rotation Forest, Decision Tree Learning

a) 한양대학교 전기전자공학부(Department of Electronics and Computer Engineering, Hanyang University)

† Corresponding Author : 정제창(Jechang Jeong)

E-mail: jjjeong@hanyang.ac.kr

Tel: +82-2-2220-4370

ORCID: <http://orcid.org/0000-0002-3759-3116>

※ “이 연구는 방위사업청 및 국방과학연구소의 재원에 의해 설립된 신호정보 특화연구센터 사업의 지원을 받아 수행되었음.”

· Manuscript received November 10, 2017; Revised November 30, 2017; Accepted December 5, 2017.

Copyright © 2017 Korean Institute of Broadcast and Media Engineers. All rights reserved.

“This is an Open-Access article distributed under the terms of the Creative Commons BY-NC-ND (<http://creativecommons.org/licenses/by-nc-nd/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited and not altered.”

1. 서론

최근 인터넷 환경이 빠르게 발달함에 따라 네트워크를 활용한 클라우드 컴퓨팅 화상회의 등 기존과는 다른 형태의 스크린 콘텐츠 (디지털 기기에서 생성된 비디오) 사용자 층이 형성되었다. 하지만 기존의 High Efficiency Video Coding (HEVC)은 카메라로 촬영된 자연영상에 맞춰 개발되었기 때문에 스크린 콘텐츠 영상을 효율적으로 압축하지 못한다는 문제가 제기되었다. 이러한 문제 때문에 2014년 1월 스크린 콘텐츠를 고려한 비디오 압축 표준 제정을 위해 ISO/IEC Moving Picture Expert Group, ITU-T Video Coding Expert Group에서 Joint Collaborate Team on Video Coding (JCT-VC)을 결성하였고 스크린 콘텐츠 확장 표준에 대한 Call for Proposal (CfP)를 공동으로 발행하였다^[1]. 스크린 콘텐츠 고유 특성을 고려하여 JCT-VC Screen Content Test Model software Ver. 8.5 (SCM 8.5^[2])가 완성되었고 스크린 콘텐츠 영상에서 기존의 HEVC 대비 55%의 BD-BR 감소를 보였다. 주요한 성능 향상의 원인으로는 현재 화면 참조 (Intra Block Copy: IBC), 색채 기반 부호화 (Palette Coding Mode: PLT), 적응적 색상 변환 (Adaptive Color Transform: ACT), 적응적 움직임 정밀도 보정 (Adaptive Motion Compensation Precision: AMCP) 등의 새로운 부호화 기술 추가가 원인이다^[3]. 하지만 막대한 성능 향상에도 불구하고 새롭게 추가된 코딩 모드들이 매 블록 사이즈마다 계산 되므로 계산량이 크게 증가하는 단점이 있다. 현재의 SCM에는 모든 Coding Unit (CU) 파티션에 대해 모든 예측 모드를 수행한 후 최적의 블록을 채택하도록 되어 있다. 또한 스크린 콘텐츠와 자연영상이 혼합된 영상을 SCM으로 부호화 할 때에는 기존의 HEVC에 비해 계산량은 증가하지만 큰 성능 향상을 얻지 못한다. 따라서 스크린 콘텐츠의 효율성을 유지하면서 빠른 부호화를 이룰 수 있는 알고리즘을 개발하는 것이 중요하다.

기존의 HEVC 고속화 연구는 많이 있었지만 HEVC Screen Content Coding (SCC) 확장 표준에 적용 가능한 고속화 연구는 거의 없다. SCC에 추가된 현재 화면 참조와 색채 기반 부호화 모드는 이전에 동일한 패턴이 출현하였을 경우 블록사이즈에 관계없이 높은 채택 확률을 보이기 때문에 CU분할 초기 결정 문제를 매우 어렵게 만들기 때문

이다^[4]. 기존의 고속화 연구는 크게 4가지 종류로 나눌 수 있다. 본 논문에서는 기존의 HEVC 고속화 연구에 대해 정리하고 스크린 콘텐츠 부호화의 고속화 연구가 필요한 이유에 대해서 설명한다.

- A. CU 조기분할 결정 방법: CU 조기분할 결정 방법은 기존의 HEVC가 모든 크기의 블록을 비교하여 최적 블록을 채택하는 과정을 고속화한 연구이다. 현재 부호화중인 블록의 이미지 복잡성을 참고하여 초기에 블록 분할을 결정하거나^[5], 주변 블록들의 크기를 참고하는 방법^[6], 블록의 Coded Block Flag(CBF)를 참고하여 초기에 분할을 결정하는 방법을 사용한다^[7]. 최근에는 Support Vector Machine (SVM)이나^[8], 인공신경망 (Neural Network: NN)^[9], 랜덤 포레스트를 이용한 방법^[10] 등의 머신러닝 분류기를 사용하여 CU 크기를 초기 결정하는 연구가 활발히 이루어지고 있다.
- B. 예측모드 수행 감소 방법: [11]에서는 화면 간 예측 부호화 과정에서 단방향 예측을 통해 양방향 예측 모드 수행을 줄인다. [12]에서 제안된 빠른 인트라 모드 알고리즘은 현재 부호화 중인 블록의 참조 샘플 유형을 분석하여 수행할 인트라 모드의 종류를 감소시킨다.
- C. 비용함수 대체 방법: Hadamard Cost를 이용하여 인트라 모드의 대략적인 결정을 통해 인트라 모드의 복잡성을 크게 감소시키는 방법이다^[13].
- D. 빠른 탐색 알고리즘: 인터모드의 움직임 탐색 (Motion Estimation) 과정에서 느슨한 탐색으로 연산의 복잡성을 줄일 수 있다. [14]에서는 블록 내 픽셀 일부만 비교하는 Partial Distortion Search (PDS) 빠른 탐색 알고리즘을 제안하였다.

이러한 기존의 HEVC 고속화 방법들은 화면 내 예측 과정과 화면 간 예측 과정의 고속화를 통해 이루어졌다. 하지만 스크린 콘텐츠는 일반적으로 적은 색상 종류, 선명한 가장자리, 반복적인 패턴, 카메라 센서 노이즈가 없으며, 급격한 영상변화가 일어나는 특징이 있다. 따라서 기존의 고속화 연구에서 사용되었던 영상의 통계적인 특성, 부호화 모

드의 선호도, 블록의 분할 구조와 화면 내 예측간의 상관관계 등의 특성을 사용하기 어렵다.

최근의 SCC 고속화 연구에는 자연영상으로 추정되는 블록의 현재 화면 참조 탐색 과정을 생략하는 방법^[15], NN분류기를 사용하여 스크린 콘텐츠를 분류하는 방법들이 연구되고 있다^[4].

본 논문에서는 회전 포레스트 (Rotation Forest) 분류기를 통해 스크린 콘텐츠 블록과 자연영상을 분류한 뒤 기존의 화면 내 예측 고속화 연구를 적용하는 방법을 사용하여 문제를 해결할 것이다. 또한 조기 CU분할 결정을 위한 부호화기의 구성과 부호화기의 학습에 사용할 블록 특징을 제안할 것이다. 제안된 스크린 콘텐츠 고속화 방법은 자연영상과 스크린 콘텐츠의 부호화 효율성을 유지하면서 불필요한 연산을 상당히 줄일 수 있음을 보여준다.

본 논문의 구성은 다음과 같다. 2장에서는 스크린 콘텐츠 부호화의 주요한 부호화 도구들을 간략하게 설명하고 고속화 연구에 미치는 영향을 논한다. 또한 기계 학습 분류 기법인 회전 포레스트 학습법에 대해 소개한다. 3장에서는 제안하는 회전 포레스트 분류기를 사용한 조기분할 결정 방법의 부호화기 구성에 대해 설명한다. 4장에서는 회전 포레스트 분류기의 분류 성능과 제안하는 부호화기의 성능을 평가한다. 마지막으로 5장에서는 결론 및 향후 연구 계획을 기술한다.

II. 기존의 기술 개요

1. 파티션 결정 방법

그림 1과 같이 스크린 콘텐츠 부호화 과정에서는 기존의 HEVC와 동일하게 쿼드트리 분할 구조를 사용한다. 64×64 크기의 Coding Tree Unit (CTU)를 시작으로 4개의 하위 CU로 분할 될 수 있으며 매 CU는 기본 코딩 단위로서 모든 예측모드가 수행된다. 부호화 순서는 상위 깊이 부호화를 진행한 후 하위 깊이의 분할 블록 부호화를 수행한다. 최종 깊이까지 진행한 후에는 하위 4개의 CU와 상위크기 CU의 Cost를 비교 후 최적의 분할 구조를 선택한다. 기존의 화면 내 예측 프레임 부호화의 경우 선형 예측에 기반한 예측

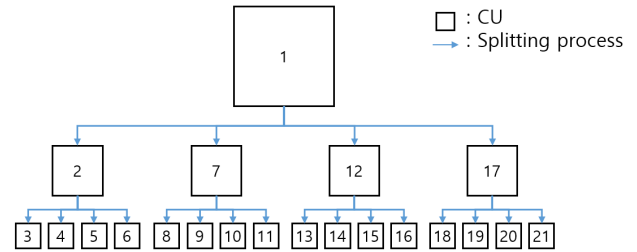


그림 1. HEVC의 쿼드트리 부호화 순서 (최대 분할 단계를 3단계로 가정)
Fig. 1. Quadtree partition coding order of HEVC (three step of maximum depth)

샘플을 생성하기 때문에 고주파 성분의 영상에서 더 작은 분할 구조의 CU가 채택되는 경향이 있다. 하지만 스크린 콘텐츠 부호화의 화면 내 예측 프레임에서는 새로운 부호화 모드의 출현으로 인해 영상의 복잡도와 관계없는 CU 분할구조가 채택된다.

2. 현재 화면 참조 부호화

스크린 콘텐츠 부호화에서는 화면 간 예측 (Inter Prediction)

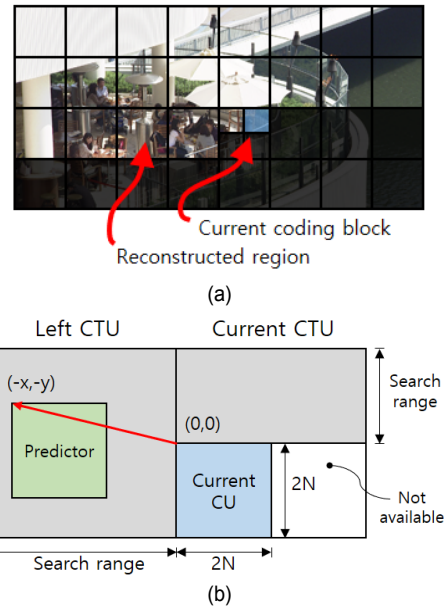


그림 2. (a) 전역적 블록 벡터 탐색의 참조 가능 영역 (b) 지역적 블록 벡터 탐색의 참조 가능 영역
Fig. 2. (a) Reference area for global block vector search (b) Reference area for local block vector search

을 현재 화면 내에서 수행하는 방법과 유사한 현재 화면 참조 모드가 추가되었다. 현재 화면 참조 기술은 그림 2와 같은 범위에서 동일한 영상의 블록을 찾아 해당 블록의 블록벡터 (Block Vector)를 기록하는 방법으로 현재 화면을 참조 가능하다. 현재 화면에 동일 패턴의 영상이 반복 될 때 높은 효율을 보이며, 기존의 화면 간 예측 모드에서 쓰인 하드웨어 구성을 재사용 가능한 장점이 있는 부호화 모드이다. 스크린 콘텐츠뿐만 아니라 기존의 자연영상에서도 동일한 패턴이 반복적으로 출현 한다면 현재 화면 참조 부호화 모드가 채택될 수 있다. 현재 화면 참조는 지역적 탐색과 전역적 탐색과정으로 탐색 과정을 수행한다. 지역적 블록벡터 탐색은 그림 2의 (b)와 같이 현재와 좌측 CTU의 복호화 된 영역을 탐색 범위로 한다. 탐색에는 다음 식 1과 같은 비용을 통해 최적의 블록 벡터를 구한다.

$$RD_Cost = SAD_{luma} + SAD_{chroma} + lamda \times BV_{bits} \quad (1)$$

지역적 블록벡터 탐색에서는 그림 2의 (a)와 같은 탐색 범위를 사용하며 CU 크기마다 부호화 과정이 다르다. 16×16 크기의 CU에서는 수직, 혹은 수평방향으로의 1픽셀 선 단위의 1차원 방향으로 탐색하는 방법으로 현재 블록과 일치하는 블록을 찾는다. 8×8 크기의 CU에서는 해쉬 기반 탐색을 사용하여 복호화 된 영상의 전체 범위를 탐색한다. 탐색 방법은 현재 부호화 중인 8×8 크기 블록의 고유한 해쉬 값을 계산하고 미리 구해 둔 복호화 된 영역의 모든 8×8블록들의 해쉬값과 일치하는 블록을 찾는 방법을 사용한다. 해쉬는 룩업테이블과 연결리스트 구조로 이루어져 있어 영상의 탐색 범위가 넓더라도 해쉬가 일치하는 탐색 대상 블록을 빠르게 찾을 수 있도록 한다. 해쉬는 블록에 대한 평균 (DC) 과 기울기로 이루어져 있으며 해쉬 (H)는 다음과 계산한다.

$$\begin{aligned} H &= MSB(DC0,3) \ll 13 + MSB(DC1,3) \\ &\ll 10 + MSB(DC2,3) \ll 7 + MSB(DC3,3) \\ &\ll 4 + MSB(Grad,4) \end{aligned} \quad (2)$$

식 (2)에서 $\ll$ 는 왼쪽방향 비트 쉬프트 연산을 의미하며, $DC1, DC2, DC3, DC4$ 는 블록을 4×4 의 하위 크기로 나눈 뒤의 평균, $Grad$ 는 블록의 기울기 합, $MSB(X,n)$ 는 X 의 최상위 비트 n 개를 취하는 함수이다.

현재 화면 참조 부호화는 블록에 대한 해쉬를 비교함으로써 현재 화면의 복호화된 전체 영역을 탐색할 수 있고 유효-왜곡 비용 비교를 통해 최적의 블록벡터를 도출한다. 현재 화면 참조 부호화는 블록벡터를 통해 적은 비용으로 화면 내에서 자주 사용되는 이미지를 예측 샘플로 생성한다. 따라서 현재 화면 참조 부호화의 채택 확률은 현재 화면의 반복적인 이미지 출현과 연관이 높은 결과를 보인다.

3. 색채 기반 부호화 (Palette Coding mode)

색채 기반 부호화는 스크린 콘텐츠 부호화에서 새롭게 추가된 부호화 모드로 블록의 사용 색상 개수가 적고 복잡한 패턴, 이전 블록에서 사용한 색상을 반복적으로 출현하는 경우 높은 부호화 효율을 보인다. 색채 기반 부호화는 현재 블록을 대표하는 색상표 (Palette)와 인덱스 맵 (Indices Map)의 조합으로 블록을 표현한다. 그림 3과 같이 색상표는 RGB 또는 YUV의 픽셀값을 색상 인덱스 (Palette Index)와 함께 저장하며 인덱스 맵에서 해당 색상의 인덱스만으로 블록을 표현한다. 하나의 블록을 부호화 할 때 색상표는 현재 CU의 색상표와 색채 예측자 (Palette Predictor)라 불리는 이전에 자주 사용된 색상표를 추가로 사용할 수 있다. 색채 예측자를 통해 유도된 색상은 색상 표현에 소모되는 추가적인 비트 소모가 적다. 따라서 색채기반 부호화 모드의 채택확률은 이전에 나왔던 색상이 반복될 경우 높아지는 결과를 보인다.

Palette	Indices Map							
RGB / YUV	0	0	0	0	0	0	0	0
0 (255, 255, 255)	0	1	1	1	0	0	0	0
1 (189, 215, 238)	0	1	1	1	0	0	0	0
2 (255, 116, 101)	0	1	2	1	0	0	0	0
	0	2	2	2	1	0	0	0
	0	2	2	2	2	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0

그림 3. 색채 기반 부호화의 색상 인덱스와 인덱스 맵

Fig. 3. Index map and Palette in Palette mode encoding

4. 회전 포레스트 학습법

회전 포레스트 학습법은 결정트리 학습법을 사용한 기계

학습 기법의 일종으로 회귀 (Regression) 분석과 분류 (Classification)의 수행이 가능하다. 결정트리는 나무 형태의 재귀적인 계층 구조로 표현되며 학습 과정을 통해 구조가 결정된다. 트리는 그림 4와 같이 노드 (Node)와 종말 노드 (Terminal Node)로 구성되어 있으며, 최상위 노드에서 분류 기준에 따라 2개의 하위 자식노드로 분할된다. 회귀 문제에서는 오차 제곱합 (Squared Sum of Error)을 사용하여 노드를 분할하며 분류 문제에서는 교차엔트로피 (Cross-Entropy) 또는 식 3의 지니 불순도 (Gini Impurity)를 통해 노드를 분할한다.

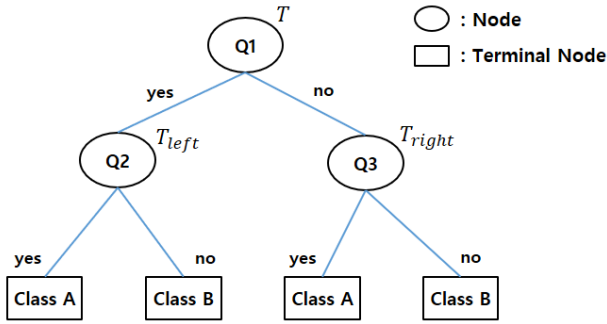


그림 4. 결정트리의 기본적인 구조
Fig. 4. A simple structure of decision tree

$$im(T) = 1 - \sum_{i=1}^M P(w_i|T)^2 = \sum_{i \neq j} P(w_i|T)P(w_j|T) \quad (3)$$

여기서 M 은 부류 (Class Label)의 전체 개수, T 는 노드 T 로 분류된 데이터의 확률을 의미하고 w_i 는 부류 i 의 확률을 의미한다. 모든 노드는 입력데이터에서 지니 불순도가 최저가 되는 분할을 찾아 2개의 자식노드로 분할된다. 최종적으로 식 4와 같이 불순도 감소가 최대가 되는 트리 구조로 학습이 이루어진다.

$$\Delta im(T) = im(T) - \frac{|X_{T_{left}}|}{|X_T|} im(T_{left}) - \frac{|X_{T_{right}}|}{|X_T|} im(T_{right}) \quad (4)$$

여기서 X_T 는 T 노드의 데이터 집합을 의미하고 T_{left} 는 현재 노드 T 의 왼쪽 자식 노드 T_{right} 는 오른쪽 자식 노드를 의미한다. 이때, 분할되지 않은 노드와 분할된 두 노드의 불순도를 비교하여 분할된 두 노드의 불순도가 더 높게

된다면 분할 과정을 중단한다. 각 노드는 지니 불순도가 최저가 되는 분할 기준을 학습 샘플 (Training Set)을 통해 결정하여 저장한다. 종말 노드에서는 검증 샘플 (Validation Set)을 통해 노드의 신뢰도를 산출하고 저장한다. 일반적으로 전체 데이터 샘플의 임의로 절반을 선택하여 학습 샘플, 나머지 절반은 검증 샘플로 사용한다. 이렇게 학습된 트리는 새로운 데이터가 입력되었을 때 저장된 분할 기준을 통해 종말노드에 도달하고 종말노드의 신뢰도를 지표로 하여 입력데이터의 분류가 이루어진다.

회전 포레스트 분류기법은 위의 결정 트리 학습법과 트리 생성 과정이 동일하지만 학습에 사용되는 특징 (Feature)을 임의로 선택하여 여러 가지 트리를 학습시킨다는 점이 다르다. 회전 포레스트는 먼저 학습에 사용되는 특징 집합에서 임의의 특징을 제외하여 특징 집합을 다시 생성한다. 임의로 나누어진 부분 집합 (Subset)에 Principal Component Analysis (PCA)를 수행하여 데이터 차원을 회전시켜 훈련을 진행한다. 이 과정을 통해 복잡한 분포의 입력 데이터에 대해 강건한 특성을 가질 수 있다^[6]. 각 트리 i 의 회전된 데이터 D_i 를 구하는 과정은 다음과 같다.

알고리즘 1

- X : 훈련 데이터 집합 (n 개의 특징과 N 개의 샘플을 갖는 $N \times n$ 행렬)
 Y : 부류 집합 ($N \times 1$ 행렬)
 L : 앙상블 모델의 개수 ($i = 1 \dots L$)
 K : 특징 집합의 부분집합 개수 (K 는 이 알고리즘의 파라미터이다.)
 F : 특징 집합
- 회전 매트릭스 R_i^a 를 준비하는 과정:
 - F 집합에서 임의의 특징 M 개를 선택하여 K 개의 부분 집합을 만든 후 F_{ij} 라 한다.
(선택과정은 독립이며, For $j = 1 \dots K$, $M = n/K$)
 - For $j = 1 \dots K$
 - F_{ij} 의 특징으로 추려낸 훈련 데이터 집합 집합을 X_{ij} 라 한다.
 - X_{ij} 의 부류를 임의로 소거한다.
 - X_{ij} 를 임의로 75%만 취득한 집합을 X'_{ij} 라 한다.
 - X'_{ij} 에 대해 PCA를 적용하고 고유값의 크기로 정렬한 고유벡터 행렬 R_i 을 얻는다.
 - R_i 를 특징 집합 F 의 특징순서대로 재정렬하여 R_i^a 라고 한다.
 - $D_i = X R_i^a$

III. 제안하는 부호화기 구성

이번 절에서는 제안하는 부호화기의 구성과 블록 흐름도를 소개하고 회전 포레스트 분류기의 특징 선택 (Feature Selection), 연관성 방법과 분류기의 학습 과정을 소개한다.

1. 제안하는 부호화기의 블록 흐름도

그림 5에서 나타낸 것과 같이 제안하는 부호화기의 흐름도는 크게 3가지 부분으로 나눌 수 있다. 특징 추출 (Feature Extraction) 부분과 분류기 1, 분류기 2 부분이다.

기존의 SCM에서는 매 CU 분할 구조마다 모든 부호화 모드를 수행한 후 최적의 분할 구조와 부호화 모드를 선택한다. 이때 대부분의 연산은 부호화 모드의 수행과정에서 발생하므로, 부호화 모드 수행 이전에 적합한 CU 분할 구조와 부호화 모드를 미리 결정해 준다면 부호화기의 복잡성을 줄일 수 있다. 따라서 제안하는 부호화기에서는 CU의 부호화 과정이 시작되면 현재 CU에 대한 특징 추출 단계를 시작한다. 특징 추출 단계에서 추출한 특징을 기반으로 적합한 부호화 모드와 CU 분할 구조를 분류기 1과 2를 통해 조기에 결정하도록 하였다.

스크린 콘텐츠 부호화에서는 새롭게 추가된 현재 화면

참조, 색채기반 부호화의 영향 때문에 영상의 복잡성과 블록 크기간의 상관성이 적다. 때문에 CU분할 구조를 조기에 판단하기가 매우 어려우므로 스크린 콘텐츠와 자연 영상을 우선적으로 분류하는 분류기 1을 배치하여 문제를 해결하였다. 이후 자연 영상으로 분류된 블록들은 분류기 2에서 조기분할 결정 문제를 풀 수 있다.

분류기 1은 현재 부호화 중인 CU를 스크린 콘텐츠 블록 (Screen Content Blocks: SCB)과 자연영상 블록 (Natural Content Blocks: NCB)으로 분류 한다. SCB는 현재 화면 참조, 색상 기반 부호화, 인트라 모드를 수행 한 후 하위 깊이의 CU 부호화 단계로 진행한다. NCB는 분류기 2로 진행하여 CU의 분할 구조를 결정한다.

분류기 2는 현재 부호화 중인 CU를 조기중단 블록 (Split-Class)과 조기중단 블록 (Unsplit-Class)으로 분류한다. S-Class의 경우 현재의 모든 RDO 과정을 생략하고 하위 깊이 CU 부호화를 진행한다. U-Class의 경우 현재 깊이의 CU 부호화만 진행 한 뒤 하위 깊이 부호화 과정을 생략하고 다음 CU 부호화를 진행한다. 분류기 2의 단계에서는 기존 HEVC의 영상 복잡성에 기반한 조기분할 결정 연구들을 사용 가능하다.

2. 특징 선택 (Feature Selection)

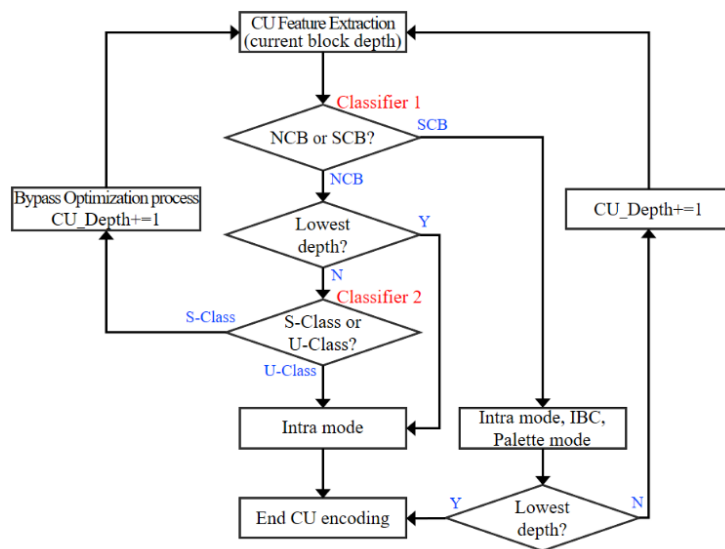


그림 5. 제안하는 부호화기의 흐름도
Fig. 5. Proposed encoder flowchart

SCB와 NCB, 그리고 S-Class와 U-Class간의 올바른 분류를 위해 블록들을 잘 표현하고 또 이들을 구분 지을 수 있는 특징을 골라 분류기의 학습에 사용하여야 한다. 본 논문에서는 연산량과 분류의 정확성을 고려하여 6가지의 특징을 선정 하였으며 다음과 같다.

A. CU Gradient: CU Gradient는 블록 내 복잡성을 나타내는 지표가 될 수 있다. CU의 Gradient 가 작을수록 CU가 분할 될 확률이 적으며 수식(6)로 나타낼 수 있다.

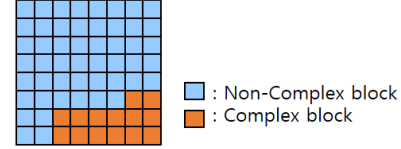
$$CU Gradient = \sum_{y=1}^8 \sum_{x=1}^8 |Y(x,y) - Y(x-1,y)| + |Y(x,y) - Y(x,y-1)| \quad (6)$$

여기서 $Y(x,y)$ 는 픽셀 x,y 지점의 밝기 값을 의미한다.

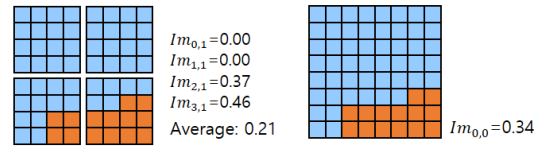
B. CU Partition Impurity: 스크린 콘텐츠 부호화에서는 새롭게 추가된 부호화 모드의 영향으로 CU의 복잡성이 균일하게 낮거나 균일하게 높은 경우 모두 CU의 분할 확률이 줄어든다. 따라서 지니 불순도를 이용하여 균일한 특성을 보이는 블록을 판단할 수 있는 특징을 제안한다. 그 계산과정은 다음과 같다. CU를 8×8의 하위 크기로 분할하여 Gradient를 구한 후 Gradient 가 미리 설정한 임계치 α (500)보다 크다면 복잡한 블록으로 간주하고 그렇지 않다면 복잡하지 않은 블록으로 간주한다. 그런 다음 그림 6의 (a)과 같이 8×8 단위로 복잡도 지도 (Complexity Map)를 만든 후 그림 6의 (b)처럼 현재 깊이의 CU 불순도와 4개로 분할 된 하위 깊이의 CU 불순도를 비교하여 분할 적합성을 알 수 있다. 예시 그림에서는 d 가 0일 때 (64×64 블록과 4개의 32×32블록의 비교) 하위 깊이로 분할 된 CU의 불순도 평균이 분할되지 않은 CU의 불순도보다 작으므로 분할 된 CU가 더 적합함을 보이는 예 이다. 식 6은 CU Partition Impurity특징의 계산 방법이다.

$$CUPartition Impurity = \frac{1}{4} \sum_{i=0}^3 Im_{i,d+1} - Im_{0,d} \quad (6)$$

$Im_{a,b}$ 는 b 깊이의 a 번째 CU의 지니 불순도를 의미하고 식 3의 불순도 계산과 동일하다. d 는 현재 CU의 깊이를 의미한다.



(a) Example of 8x8 complexity map of 64x64 CU



(b) Example of partition impurity
(0.21 < 0.34 this CU is more likely to be partition)

그림 6. (a) CU파티션 불순도 맵 (b) 분할 불순도 비교의 예
Fig. 6. (a) CU partition impurity map (b) Example of partition impurity

C. Intra Mode Cost: 스크린 콘텐츠 부호화 과정에서 화면 내 예측 모드의 복잡성은 상대적으로 낮다. 수직, 수평 방향의 평탄한 패턴을 띄는 스크린 콘텐츠 블록은 Angular Mode로 부호화 할 경우 부호화 비용이 매우 적은 특징이 있으므로 추가적인 스크린 콘텐츠 부호화 모드의 수행 여부를 판단하는데 좋은 지표가 될 수 있다. 식 7과 같이 35가지 방향의 예측 p 에 대해 부호화를 수행 한 후 최적의 비용을 나타낸다.

$$Intra Mode Cost = \min_p (SATD_p + \lambda_p \times B) \quad (7)$$

D. CU Color Count: 현재 CU의 RGB또는 YUV 색상 구성 요소 개수를 계산한다. 색상 기반 부호화의 채택과 연관이 높다.

E. Zero Gradient Count: Gradient가 0인 픽셀의 개수를 계산한다. 스크린 콘텐츠는 카메라 센서 노이즈가 없기 때문에 Zero Gradient Count가 높게 측정되므로 스크린 콘텐츠를 판단하는 지표가 될 수 있다.

F. Gradient Peak: 현재 CU의 내 블록 픽셀 Gradient 중 최댓값을 나타낸다. 글자 등 주파수 변환이 어려운 날

$$Gradient\ Peak = \max_{x,y \in CU} (|Y(x,y) - Y(x-1,y)| + |Y(x,y) - Y(x,y-1)|) \quad (8)$$

카로운 이미지를 판단할 수 있다. 식 8과 같은 과정으로 계산한다.

$$\mu_j(x) = \frac{1}{L} \sum_{i=1}^L d_{ij}(xR_i^a) \quad (j=1, \dots, c) \quad (9)$$

3. 연판정 (Soft Decision) 방법

각 분류에 대해 학습된 결정트리의 종말 노드 신뢰도가 임계치 β (85%)보다 낮을 경우 블록의 분류 판단을 보류하는 연판정 방법을 사용한다. 그림 7은 분류기 2의 연판정 방법을 보인 예이다. 학습된 트리는 앙상블 모델로 여러 개의 트리에서 여러 개의 결과를 갖기 때문에 각 트리가 갖는 결과의 평균을 취하는 방법을 사용하여 최종 신뢰도를 계산하였다. 그림 7의 최종 출력은 약 0.13으로 S-Class에 대한 신뢰도는 약 86%에 해당한다. 신뢰할 수 있는 임계치 범위에 속하므로 S-Class로 분류된다. 입력 데이터를 x , 부류의 개수를 c , $d_{ij}(xR_i^a)$ 를 분류기 D_i 에 의해 결정된 노드 신뢰도라고 하였을 때, 최종 신뢰도는 다음 식으로 계산된다.

분류기 1에서 연판정 블록으로 판단 될 경우에도 현재 블록 크기의 모든 예측 모드를 수행 한 후 하위 깊이 CU 부호화를 진행한다. 분류기 2에서 연판정 블록으로 판단 될 경우 현재 블록 크기의 모든 예측 모드를 수행 한 후 하위 깊이 CU 부호화를 진행한다.

4. 분류기 선택

제안하는 흐름도의 분류기 위치에는 다양한 모델의 분류기가 선택 될 수 있다. Logistic Regression, Linear Perceptron Classifier, NN, SVM, 결정트리 학습법 등을 사용 가능 하지만 스크린 콘텐츠 분류 문제는 선형 분류로 해결 가능 하지 않기 때문에 NN이나 커널법을 사용한 SVM, 결정트리 학습법 등을 사용한 기계학습 분류기를 사용하는 것이 적합하다^[4]. 본 논문에서는 분류기의 결정트리 학습법에 앙상블 모델을 적용한 로테이션 포레스트 분류기를 사용하였다.

5. 분류기 학습 방법

현재 부호화중인 CU의 부호화 모드와 분할을 조기에 추정하기 위하여 지도학습 (Supervised learning) 방법을 사용한다. 스크린 콘텐츠 공통 테스트 조건 (Common Testing Condition: CTC)^[17]의 각 시퀀스 (Sequence) 별로 길이를 균등하게 나누어 10픽처를 취득한 뒤 기존의 SCM을 사용하여 부호화 하였다. 부호화된 CU들은 특징을 추출하고 최종적인 부호화 모드와 분할 구조를 함께 기록하여 훈련 샘플을 만든다. 만들어진 훈련 샘플은 회전 포레스트 분류기의 학습에 사용한다. CU 크기에 따라 특징들의 스케일이 다르고 QP에 따라 부호화 모드의 선호도가 다르기 때문에 각 CU 크기별, QP별로 분류기를 다르게 사용한다. 훈련 샘플은 전체 데이터 샘플의 임의의 절반으로 구성하고 검증 샘플은 나머지 절반으로 한다. 추가로 종말 노드의 샘플 개

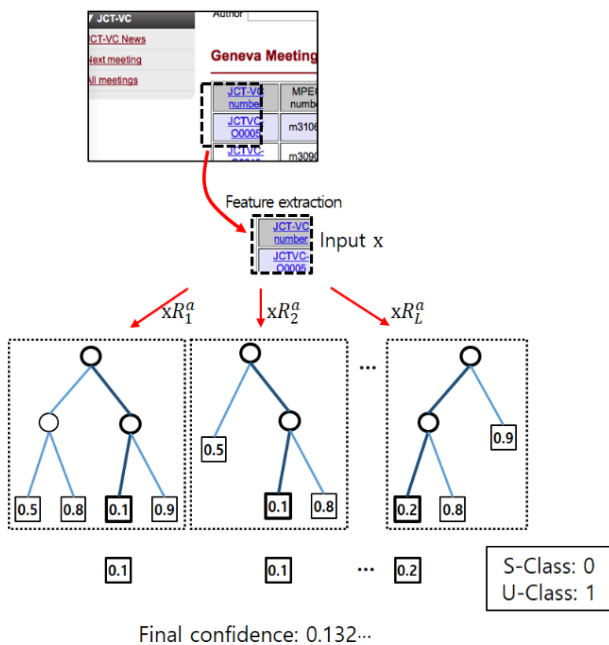


그림 7. 회전 포레스트 분류기의 연판정 방법
Fig. 7. Soft-decision of rotation forest classifier

수가 전체 학습 샘플 수의 0.5% 이하가 되었을 때 노드 분할을 중단하도록 분할 중단 조건을 설정 하였다.

IV. 실험결과 및 분석

본 논문에서 제안하는 스크린 콘텐츠 조기분할 결정 방법을 평가하기 위해 All Intra 프로파일의 스크린 콘텐츠 공통 테스트 조건 CTC를 사용하여 실험하였고, 테스트 소프트웨어는 HM-16.15 SCM-8.5^[2], 실험환경은 Intel Xeon E5-2690 @ 3.00GHz 프로세서 환경에서 실험하였다. 객관적 부호화 성능 평가를 위해 BjØntegaard-Delta 측정 방법을 사용하였고^[18], 평균 시간 감소율 ATS (Average Time Saving)는 다음 수식을 사용한다.

$$ATS = \frac{Enc.time(anchor) - Enc.time(proposed)}{Enc.time(anchor)} \times 100(\%) \quad (10)$$

Rotation Forest 분류기의 학습에 사용한 소프트웨어는 통계프로그래밍 언어 R의 Recursive Partitioning and Regression Trees (Rpart) library (Ver. 4.1-11)를 사용하였다. 임계치 사용자 정의 파라미터 α 와 β 는 예비 실험을 통해 500과 85%로 결정하여 사용하였다.

실험결과 제안하는 회전 포레스트 분류기의 계산 복잡성

표 1. 학습된 회전 포레스트 분류기의 분류 정확도

Table 1. Classification accuracy of Rotation Forest Classifier

			CU Depth			
			64×64	32×32	16×16	8×8
Classifier 1	QP	22	87.3	90.9	78.2	81.3
		27	88.2	90.5	80.3	81.9
		32	89.9	90.6	80.0	83.6
		37	88.8	89.8	79.7	84.6
Classifier 2	QP	22	93.6	84.7	77.5	*
		27	93.5	84.7	80.1	*
		32	91.6	84.6	82.1	*
		37	90.5	84.3	82.5	*

*8×8 block is lowest depth CU that is always pruned

은 전체 인코딩 과정의 0.4%를 차지했으며 분류기의 분류 정확도는 표 1과 같았다. QP별로 선호되는 예측 모드가 다르기 때문에 약간의 분류 정확도 차이를 보인다. 분류기 1의 8×8크기의 블록과 분류기 2의 16×16블록의 결과를 보면 QP가 낮아질 때 마다 정확도가 낮아지는 결과를 보인다. 이는 QP가 낮아질수록 분류 정확도가 낮은 해쉬기반 전역적 블록벡터 탐색의 채택확률이 높아졌기 때문이다. 해쉬기반 전역적 블록벡터 탐색의 분류 정확도가 낮은 이유는 주어진 6개의 특징 (Feature)을 통해 화면 내 중복 블록의 존재 여부를 알기 어렵기 때문인 것으로 보인다.

표 2는 제안하는 회전 포레스트 분류기를 사용한 스크린 콘텐츠 공통 실험 조건 영상의 부호화 성능을 나타낸 표이

표 2. 제안하는 조기분할 방법의 실험 결과

Table 2. Experimental results of the proposed fast partition decision method

Type	Sequences	Resolution	BD-BR	BD-PSNR	ATS(%)
CC	Kimono	1080p	1.051	-0.021	79.9
M	MissionControlClip3	1080p	2.884	-0.286	41.5
TGM	Console	1080p	3.466	-1.031	34.3
TGM	Desktop	1080p	3.651	-1.197	41.9
TGM	FlyingGraphics	1080p	1.800	-0.271	17.2
TGM	Map	720p	2.119	-0.220	42.0
TGM	Programming	720p	3.059	-0.327	32.1
A	Robot	720p	2.022	-0.084	62.3
TGM	SlideShow	720p	6.795	-0.552	40.0
TGM	Web_browsing	720p	2.553	-0.463	25.0
M	BasketballScreen	1440p	4.399	-0.393	45.4
M	MissionControl2	1440p	3.530	-0.311	46.7
Average			3.111	-0.430	42.4

TGM: Text and Graphics with Motion

MC: Mixed Content

A: Animation

CC: Camera-Captured Content

다. Kimono 영상에서 가장 좋은 성능을 보였으며, SlideShow 영상에서 가장 좋지 않은 결과를 보였다. Kimono 영상은 대부분이 자연영상으로 이루어져 있고 8×8블록의 사용이 거의 없는 영상이다. 때문에 분류기 1에서 높은 정확도로 NCB로 분류되어 스크린 콘텐츠 부호화의 모드 사용을 줄일 수 있어 가장 비약적인 부호화 속도 향상을 얻을 수 있었다. 반면 SlideShow 영상은 모서리가 날카롭지만 블록에 사용된 색상이 다양한 영상이기 때문에 SCB로 올바르게 분류되지 못한 결과를 보였다. 이는 분류기의 학습에 사용된 공통 테스트 조건 영상중 SlideShow 영상이 가장 특이한 형태의 영상이기 때문이다. 이 문제를 해결하기 위해서는 CTC 영상을 균등하게 분할하여 학습 샘플을 만들었던 방법을 변경하여 문제가 되는 특이영상을 학습샘플로 선정해야 할 것으로 보인다. 이외에도 Console, Desktop 영상 또한 BD-BR증가가 높았다. 이 두 영상은 영상 내 중복 패턴이 화면 전체 영역에서 나타나는 특징을 보이는 영상이다. 전역적 탐색을 사용하는 8×8 크기 블록의 분류 정확도가 높지 못한 것이 성능 하락의 원인이다.

표 3은 기존의 다른 스크린 콘텐츠 고속화 방법과의 비교이다. 제안하는 방법은 약간의 BD-BR 증가를 보였지만 부호화 속도 면에서 큰 향상을 보였다.

표 3. 제안하는 조기분할 방법의 실험 결과

Table 3. Experimental results of the proposed fast partition decision method

Previous Fast Encoder works	Codebase	BD-BR	ATS(%)
Tsang, Chan and Siu [12]	SCM-2.0	0.66	29%
Duanmu, Ma, Wang [4]	SCM-4.0	3.69	36%
Proposed algorithm	SCM-8.5	3.11	42%

V. 결론 및 향후 연구 계획

본 논문에서는 기계학습 분류 기법인 회전 포레스트 분류기를 사용한 부호화 방법을 제안하였다. 기존의 조기분할 결정 방법들은 HEVC 화면 내 부호화과정의 CU 분할 특성에 맞춰 연구되었기 때문에 스크린 콘텐츠 부호화의 새로운 도구들을 사용할 때 올바른 조기분할 결정을 내리지 못한다. 제안하는 방법은 스크린 콘텐츠 영상과 자연영상을 먼저 분류한 다음 블록 조기분할을 결정하는 방법을

사용하여 기존의 조기분할 결정 알고리즘들이 스크린 콘텐츠에 적용되기 어려웠던 점을 해결하였고 이에 적합한 부호화기 흐름도를 제시하였다. 또한 분류기의 학습에 CU 분할 불순도라는 새로운 특징을 제시하였으며, 회전 포레스트 분류 알고리즘을 사용하여 비전형적인 분류문제를 풀 수 있으면서도 스크린 콘텐츠 부호화 과정의 계산 복잡성을 증가시키지 않음을 보였다. 제안하는 방법은 스크린 콘텐츠 공통 실험 조건에서 3.11%의 BD-BR 증가와 42%의 부호화 시간 감소를 이루었다. 실험 결과 자연 영상에서 높은 성능을 보였으며, 스크린 콘텐츠 영상이지만 사용색상이 많거나 화면 내 중복이 많은 영상에서 낮은 성능을 보였다. 향후 화면 내 패턴 중복에 대한 블록 특징을 추가하여 분류기를 학습한다면 보다 큰 성능 향상을 얻을 수 있을 것으로 보인다.

참 고 문 헌 (References)

- [1] ISO/IEC JTC1/SC29/WG11 and ITU-T Q6/SG16, MPEG2014/N14175/VCEGAW90, Joint Call for Proposals for Coding of Screen Content, San Jose, USA, Jan. 2014.
- [2] HM-16.16+SCM-8.5 Software, <https://hevc.hhi.fraunhofer.de/trac/hevc/browser/tags/HM-16.16%2BSCM-8.5> (accessed Nov. 01, 2016).
- [3] Joshi, J. Xu, R. Cohen, S. Liu, Y. Ye (editors) "High Efficiency Video Coding (HEVC) Range Extensions text specification: Draft 7, Document JCTVC-Q1005," in ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, Apr. 2014.
- [4] F. Duanmu, Z. Ma, and Y. Wang, "Fast CU partition decision using machine learning for screen content compression," in IEEE Int. Conf. Image Process. (ICIP), pp.4972 - 4976, Sep. 2015.
- [5] J. Jang, H. Choi, and J. Kim, "Fast PU Decision Method Using Coding Information of Co-Located Sub-CU in Upper Depth for HEVC," Journal of Broadcast Engineering, Vol.20, No.2, pp.340-347, Mar 2015.
- [6] D. Lee, and J. Jeong, "Fast intra coding unit decision for high efficiency video coding based on statistical information," Elsevier Signal Processing Image Communication Vol. 55, pp. 121-129, July. 2017.
- [7] S. Jeon, N. Kim, and B. Jeon, "CU Depth Decision Based on FAST Corner Detection for HEVC Intra Prediction," Journal of Broadcast Engineering, Vol.21, No.4, pp.484-492, July 2016.
- [8] Y. Zhang, S. Kwong, L. Xu, and G. Jiang, "DIRECT mode early decision optimization based on rate distortion cost property and interview correlation," IEEE Trans. Broadcast, vol. 59, no. 2, pp. 390 - 398, Jun. 2013.
- [9] J. Chiang, W. Chen, L. Liu, K. Hsu, and W. Lie, "A fast H.264/AVC based stereo video encoding algorithm based on hierarchical two-stage

- neural classification,” IEEE Signal Process, vol. 5, no. 2, pp. 309 - 320, Apr. 2011.
- [10] S. Ryu and J. Kang, “Machine-Learning based Fast Intra Mode Decision Algorithm in HEVC,” International Technical Conf. on Circuits/Systems, Computers and Communication (ICT-CSCC), 2017.
- [11] W. Han, J. Ahn, J. Lee, “Early Decision of Inter-prediction Modes in HEVC Encoder,” Journal of Broadcast Engineering, Vol.20, No.1, pp.171-182, Jan 2015.
- [12] S. Tsang, Y. Chan, and W. Siu, “Fast and efficient intra coding techniques for smooth regions in screen content coding based on boundary prediction samples,” in Proc. ICASSP, pp. 1409 - 1413, 2015.
- [13] Y. Piao, J. Min, and J. Chen, Encoder Improvement of Unified Intra Prediction, document JCTVC-C207, Jan. 2013.
- [14] C. Bei and R. Gray, “An improvement of the minimum distortion encoding algorithm for vector quantization,” IEEE Trans. Commun, vol. COM-33, pp. 1132 - 1133, Oct. 1985.
- [15] D. K. Kwon and M. Budagavi, “Fast intra block copy (IntraBC) search for HEVC screen content coding,” in IEEE International Symposium on Circuits and Systems (ISCAS), Melbourne VIC, 2014, pp. 9-12.
- [16] J. Rodriguez, L. Kuncheva and C. Alonso, “Rotation Forest: A New Classifier Ensemble Method,” in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 10, pp. 1619-1630, Oct. 2006.
- [17] H. Yu, R. Cohen, K. Rapaka, and J. Xu, Common Test Conditions for Screen Content Coding, document JCTVC-T1015, Feb. 2015.
- [18] G. Bjontegaard, “Calculation of average PSNR differences between RD curves,” Video Coding Experts Group (VCEG), VCEG-M33, Austin, Texas, U.S.A., April, 2001.

저 자 소 개



허 정 환

- 2015년 8월 : 한경대학교 전자공학과 졸업 (학사)
- 2015년 8월 ~ 현재 : 한양대학교 전자컴퓨터통신공학과 (석사과정)
- ORCID : <https://orcid.org/0000-0001-8670-9800>
- 주관심분야 : 고효율 영상 부호화 기술 (HEVC), 영상처리, 정보융합



정 제 창

- 1980년 2월 : 서울대학교 전자공학과 (학사)
- 1982년 2월 : KAIST 전기전자공학과 (석사)
- 1990년 : 미국 미시간대학 전기공학과 (공학박사)
- 1980년 ~ 1986년 : KBS 기술연구소 연구원 (디지털 및 뉴미디어 연구)
- 1990년 ~ 1991년 : 미국 미시간대학 전기공학과 연구교수 (영상 및 신호처리 연구)
- 1995년 ~ 현재 : 한양대학교 전자컴퓨터통신공학과 교수 (영상통신 및 신호처리 연구실)
- 1990년 12월 : 정보통신부장관상 수상
- 1998년 11월 : 과학기술자상 수상
- 2007년 : IEEE Chester Sall Award 수상
- 2008년 : ETRI Journal Paper Award 수상
- 2011년 5월 : 제 46회 발명의 날 독조근정훈장 수훈
- ORCID : <http://orcid.org/0000-0002-3759-3116>
- 주관심분야 : 영상처리, 영상압축, 3DTV