

KCC2017

SW 구현/데모 경진대회

ARM SoC 기반의 HMD를 위한 HEVC 병렬처리 최적화 기법

가천대학교 일반대학원
컴퓨터공학 전공 석사과정
류영일

1. Motivation

2014년 베를린에서 열린 IFA2014에서는 삼성전자가 오큘러스와 협력하여 개발한 기어 VR을 발표하며 VR시장에 진출하였고, 구글은 누구나 손쉽게 VR 및 360° 파노라마 뷰 콘텐츠를 즐길 수 있도록 제작된 HMD (Head-mounted Display)인 카드보드를 출시하였다. 최근 페이스북 또한 17개의 카메라가 내장되어 있는 360° 파노라마 뷰 촬영용 카메라인 FACEBOOK SURROUND 360을 선보였다. 이처럼 세계의 여러 IT 기업 및 전문가들은 HMD와 360° 파노라마 뷰를 제공하는 카메라 및 콘텐츠에 이목을 집중하고 있다.

HMD는 영상을 출력하는 디스플레이가 사용자의 눈에 가까운 디바이스의 특성상 고화질의 영상이 필요하고, 360° 파노라마 뷰를 지원하기 위한 큰 크기의 영상을 처리해야만 한다. 때문에 기존 영상 콘텐츠에 비하여 HMD를 위한 영상의 데이터량은 매우 증가하였고, 사용자의 움직임을 추적하며, 딜레이 없는 영상 서비스를 지원하기 위한 연산 복잡도 또한 증가하였다.

한편, 4K, 8K 영상과 같은 UHD (Ultra High-Definition) 영상에 대한 사용자들의 요구에 발맞춰 ITU-T VCEG (Video Coding Experts Group)과 ISO/IEC MPEG (Moving Picture Group)이 공동으로 조직한 JCT-VC (Joint Collaborative Team on Video Coding)는 2013년에 차세대 비디오 부호화 기술인 HEVC (High Efficiency Video Coding)를 표준화 하였다. 이어서 JVET (Joint Video Exploration Team)는 360° 비디오를 본격적으로 지원하기 위한 표준인 MPEG-I (Immersive media)를 2017년 현재 논의 중이다.

본 제안서는 ARM SoC 기반의 HMD 시스템 상에서 증가한 비디오 디코딩 복잡도를 지원하기 위하여 새로운 방식의 HEVC 기반 비디오 병렬처리 기법을 제안한다.



그림 1. 구글의 카드보드와 삼성전자의 기어 VR



그림 2. 파노라마 뷰의 예시

2. Background

2.1 JCT-VC HEVC 비디오 표준

HEVC는 ITU-T VCEG과 ISO/IEC MPEG이 공동으로 조직한 JCT-VC에서 2013년에 표준화한 최신 비디오 코딩 표준이다. HEVC는 기존의 주요 비디오 코딩 표준들과 유사하게 블록 기반의 비디오 부호화 방식을 사용하지만, 4K, 8K와 같은 UHD 영상을 지원하기 위하여 향상된 비디오 압축 기술을 제공한다. HEVC는 기존 H.264/MPEG-2 AVC에 비하여 대형화면 기준 약 50% 향상된 압축효율을 내며, Tile과 WPP (Wavefront Parallel Processing)와 같은 새로운 병렬처리 기술들이 추가되었다.

- 1) Tile: 프레임을 하나 이상의 사각형 영역으로 분할하여, 각 영역을 독립적으로 인코딩 및 디코딩 하는 신택스 (Syntax)이다. 타일은 각 스레드 (Thread) 간의 복잡한 동기화 이슈 없이 병렬처리가 가능하며 슬라이스 기법에 비해 압축효율이 조금 더 높다.
- 2) WPP: 프레임을 구성하는 CTU (Coding Tree Unit) 블록들이 행 (Row) 단위로 병렬처리가 가능하도록 정의된 신택스이다. 현재 CTU 행의 CTU 두 개가 처리된 뒤에 다음 행의 CTU들이 곧바로 처리될 수 있도록 하였는데, 이는 화면내압축 (Intra-Picture Coding)의 각 CU (Coding Unit) 처리 시, 화면의 상단과 좌측의 코딩 정보를 이용하는 점을 고려하였기 때문이다.

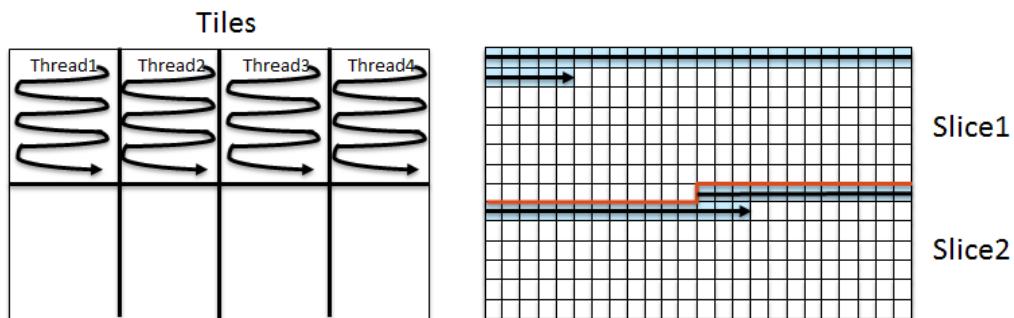


그림 3. Tile과 WPP를 활용한 병렬처리의 예시

2.2 ARM big.LITTLE Architecture

최근 ARM 계열 CPU에는 처리해야 할 작업량에 따라서 각각 서로 다른 처리능력을 갖는 Big 코어와 Little코어에 작업을 효율적으로 할당해 전력 낭비를 줄이고, 성능을 향상시키는 big.LITTLE 기술이 적용되고 있다. 서로 다른 성능을 갖는 하드웨어로 구성된 시스템에서는 특정 스레드를 어느 코어에 할당해야 하는지 결정하고, 멀티미디어 응용 서비스 제공 시 연산 복잡도가 서로 다른 비디오 처리 작업을 최적의 비대칭 코어에 할당하는 기법이 필요하다.

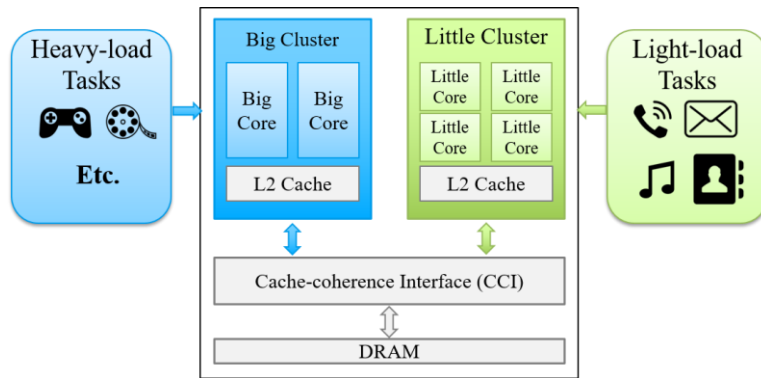


그림 4. big.LITTLE 구조의 비대칭 멀티코어 CPU의 예시

2.3. 360° Immersive media를 위한 Projection 기법

일반적으로 360° Immersive media의 경우, 구 (Sphere) 형상으로 영상을 렌더링하여 사용자에게 제공하며, 마치 영상 속에 위치한 것과 같은 몰입감을 사용자에게 제공한다. 하지만 이러한 구 형상의 영상은 기존 사각형 형상의 비디오를 위한 인/디코더에서의 처리가 제한적이다. 때문에 구 형상의 영상을 사각형 형상의 영상으로 프로젝션 (Projection) 해야 할 필요가 있다. 이를 지원하기 위하여 여러 프로젝션 기법들이 제안되었으며, 그 중 가장 보편적으로 활용되는 기법으로는 ERP (Equirectangular projection) 및 CMP (Cube Map Projection)이 있다. 이러한 프로젝션 기법들은 다양한 카메라로부터 취득된 영상들을 하나의 영상으로 맵핑을 하게 되는데, CMP의 경우 사용자가 현재 바라보고 있는 정면 영상과 그 외 방향의 영상 크기를 다르게 하여 프로젝션 한다.

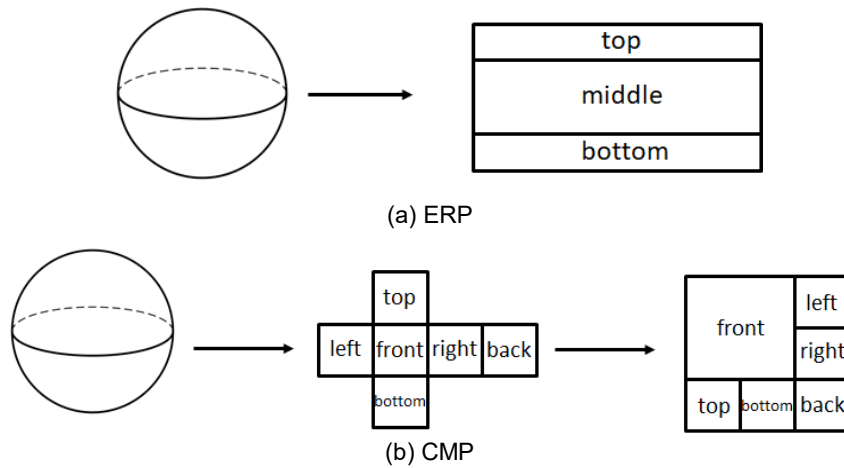


그림 5. 최근 제안되고 있는 360° 비디오를 위한 프로젝션 기법

3. Proposed Method

HMD를 위한 360° 파노라마 뷰 영상은 영상의 화질과 해상도 (Resolution)가 기존 FHD (Full High-Definition) 해상도 영상에 비하여 매우 높기 때문에 디코딩 복잡도 또한 매우 높다. 이렇게 높은 디코딩 복잡도로 인하여 디코딩 속도가 사용자의 움직임에 따른 시점 변화 속도를 따라가지

못하는 시점 변화 지연 현상을 개선하기 위하여, 본 제안서는 ARM SoC 기반의 HMD를 위한 360° 파노라마 뷰 비디오 병렬 디코딩 최적화 기법을 제안한다. 제안하는 기법은 HEVC Tile-level Parallel Processing 기법에 기반한다.

3.1 균등 (Uniform) 타일 분할 방식의 문제점

그림 6은 그림 7의 (a)와 같이 프레임을 균등하게 6개의 타일로 분할한 PeopleOnStreet 시퀀스의 타일 별 평균 디코딩 시간을 보여주는 그래프이다. 그림 6에서 볼 수 있듯이 비대칭 멀티코어 환경에서 균등하게 분할된 타일을 활용하여 비디오 병렬 디코딩을 진행하면, Big core에서 디코딩이 진행된 타일 1, 2와 Little core에서 디코딩이 진행된 타일 3, 4, 5, 6의 평균 디코딩 타임이 균등한 것이 아니라 상당한 차이가 발생하게 된다. 이는 각 타일의 디코딩 타임이 해당 타일 각각의 연산 복잡도와 스레드가 할당된 CPU 코어의 처리능력에 영향을 받기 때문이다. 이러한 현상은 디코딩 타임이 가장 짧은 타일이 먼저 디코딩 완료되어도 디코딩 타임이 가장 긴 타일의 디코딩이 완료되기를 기다려야 하는 상황을 발생시키며, 병렬처리 성능을 저하시키게 된다.

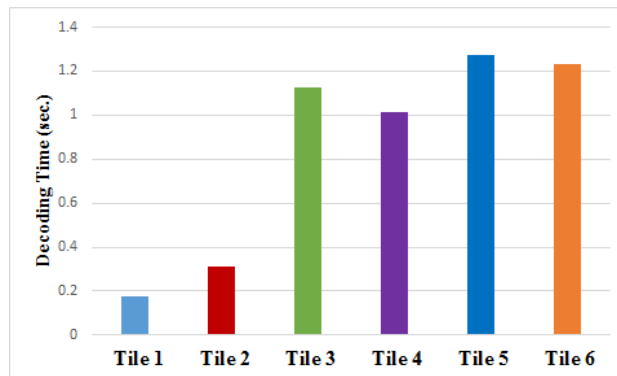


그림 6. 균등한 6개의 타일로 분할된 PeopleOnStreet의 타일 별 평균 디코딩 타임



(a) 균등한 크기의 타일 분할 (b) 각 코어의 처리능력을 고려한 타일 분할

그림 7. 서로 다른 방법으로 6 개의 타일로 분할된 PeopleOnStreet의 모습

3.2. Proposed Method: ARM SoC 기반의 HMD를 위한 HEVC 병렬처리 최적화 기법

본 연구가 제안하는 타일 분할 방식은 사전 연구를 통하여 산출된 영상의 해상도와 비디오 디코딩 연산 복잡도 사이의 상관관계를 나타내는 다중회귀분석 모델을 활용하여 다음과 같이 진행한다. 그 후, 그림 8과 같은 과정을 거쳐 Big코어와 Little코어의 성능 비에 기반하여 각 코어에 할당할 타일의 디코딩 연산 복잡도 비를 산출하고, 해당 연산 복잡도 비율에 매핑(Mapping)되는 크기의 타일을 각 코어에 할당한다.

이와 같은 방식으로 타일을 분할하면 그림 7의 (b)와 같은 형태로 타일이 비 균등하게 분할되며, 그림 7의 (b)의 경우, 크기가 큰 2개의 타일은 2개의 Big 코어에 각각 할당되며, 크기가 작은 4개의 타일은 4개의 Little 코어에 각각 할당되게 된다. 그 결과, 모든 타일의 디코딩 타임이 거의 비슷하게 평준화되기 때문에 CPU 코어들의 성능을 고려하지 않고 균등하게 타일을 분할하였을 때보다 병렬처리 효율이 향상되며, 그림 9는 그 예시를 보여준다.

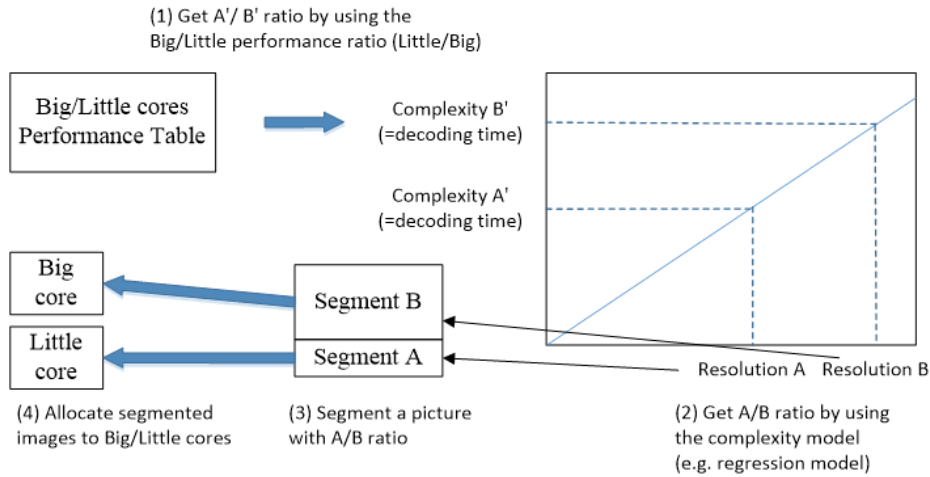
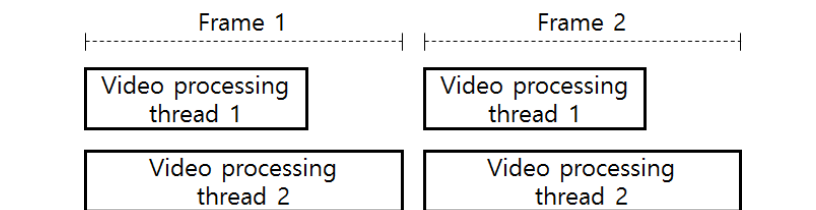
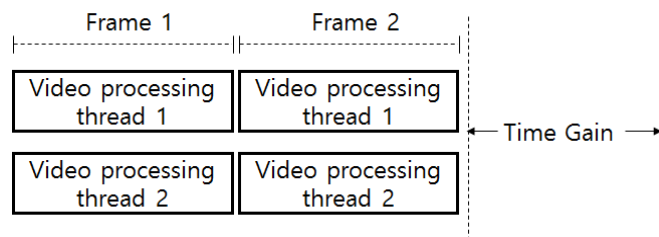


그림 8. 제안하는 비대칭 멀티코어 처리능력 기반 타일 분할 방식



(a) 비대칭 코어의 처리능력을 고려하지 않은 경우의 디코딩 타임



(b) 코어의 처리능력 기반 타일분할 방식을 적용한 경우의 디코딩 타임

그림 9. 균등 타일 분할과 제안하는 타일 분할 방식의 비교

4. Demonstration

제안하는 방식은 비디오 영상을 여러 개의 비 균등한 크기의 타일로 분할하고, 각 분할된 타일을 Big 코어와 Little 코어에 할당한다. 그림 10은 제안하는 방식의 HEVC 인/디코딩 구조를 블록 다이어그램으로 설명한다.

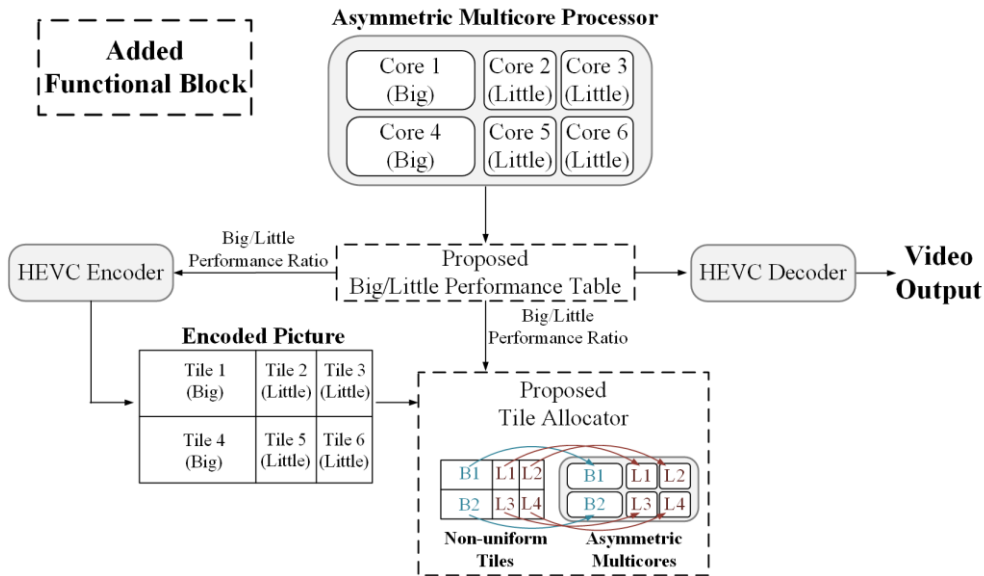


그림 10. 제안하는 HEVC 인/디코딩 시스템의 블록 다이어그램
(2 Big core와 4 Little core를 고려한 경우)

본 데모에서는 두 개의 삼성 갤럭시 S7 엣지 안드로이드 스마트폰을 활용하여 Side-by-side로 제안하는 방식과 기존 방식의 성능 차이를 비교하여 설명한다. 삼성 갤럭시 S7 엣지 스마트폰의 프로세서는 비대칭 멀티코어 아키텍처에 기반하여 설계되었으며, 본래 4개의 Big코어와 4개의 Little코어를 가지고 있으나, 배터리 효율을 위하여 2개의 Big코어는 평상시에는 오프라인 상태로 유지된다. 그러므로 본 데모는 2개의 Big코어와 4개의 Little코어만을 고려하여 진행한다.

그림 11은 본 데모 진행을 위해 구현된 비디오 디코더를 삼성 갤럭시 S7 엣지 스마트폰에서 구동 시킨 화면이며, 그림 12는 Side-by-side 데모의 진행 시나리오를 보여준다. 해당 데모 시나리오, 4K UHD 테스트시퀀스(*PeopleOnStreet*, *Traffic*) 그리고 JCT-VC의 CTC (Common Test Condition)을 적용하여 실험을 진행하였을 때에는 그림 13 (a), (b)와 같이 평균 25%의 성능 향상을 확인하였다.



그림 11. 제안된 비디오 디코더의 구동화면

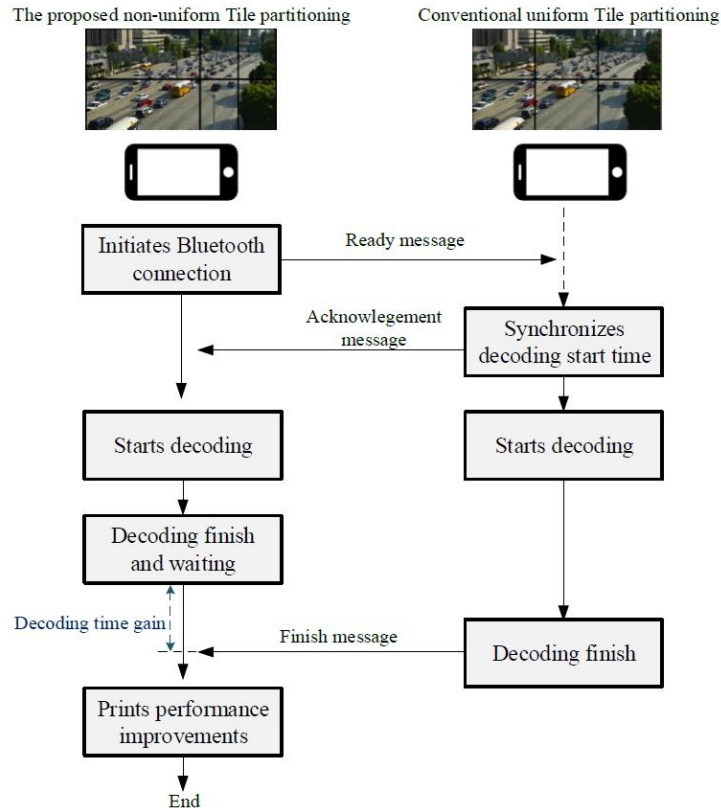


그림 12. Side-by-side 데모 시나리오

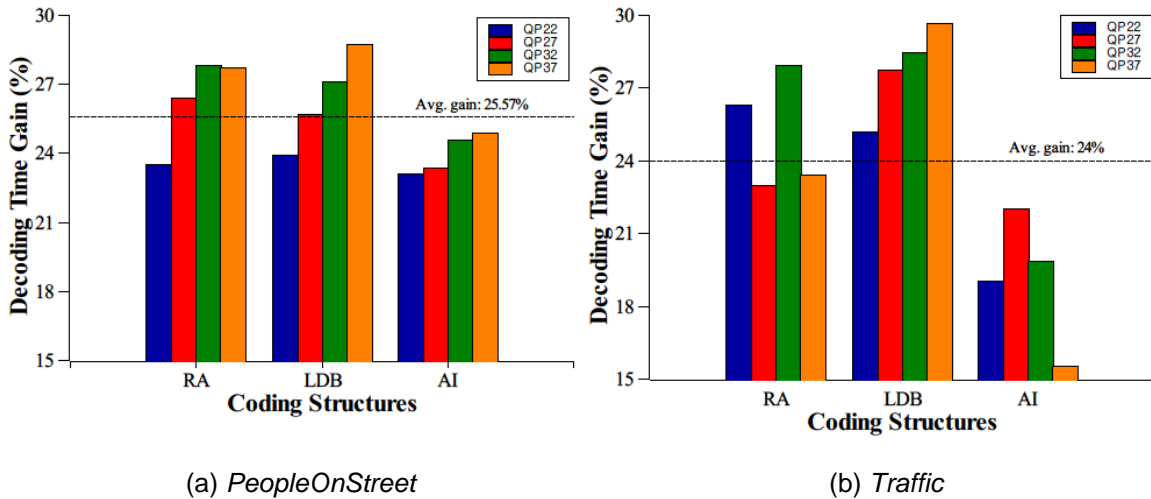


그림 13. 4K UHD 영상을 대상으로 한 실험결과

4. Conclusion

본 제안서는 비대칭 멀티코어 시스템 상에서 HEVC 병렬 디코딩 최적화를 위하여, 각 CPU 코어 각각의 처리능력에 적합한 디코딩 연산 복잡도를 갖는 크기로 타일들을 분할하여 해당 코어에 할당하는 방식을 제안하였다. 제안하는 방식은 삼성 갤럭시 S7 엣지 스마트폰을 위하여 구현되었으며, Side-by-side로 기존 방식과 성능 비교를 위한 데모를 진행한다. 기존 방식인 각 코어의 처

리능력을 고려하지 않고 균등한 크기의 타일로 프레임을 분할하는 방식과 비교 실험하였을 때, 제안하는 방식을 사용하였을 때 평균 25%의 성능 향상이 발생함을 확인하였다.