* 모든 컬러 값들을 grayscale로 변환해주는 공식에 대입하였다. 이는 특정 컬러 값이 많이 나타나는 이미지에서 불필요하게 동일한 연산을 반복하는 결과를 불러온다. 이러한 문제를 해결할 수 있는 효율적인 컬러 변환 방법에 대하여 기술한다.
* 컬러는 한 픽셀당 3개의 값으로 표기되고, 각 컬러는 2의 8승까지의 수를 갖는다. 따라서, (2의 8승)3승만큼의 엔트리로 구성된 테이블을 만들 수만 있다면, 논리적으로 어떠한 값도 저장한 다음 즉시 사용이 가능하다. 예를 들어, 하나의 픽셀의 rgb값을 얻게 될 경우, 이 rgb의 grayscale계산식에 사용되는 값이 없다면 이를 저장해 테이블에 넣어두고, 만일 있다면 이를 즉시 가져오기만 하면 되니, 이미 본 적이 있는 rgb의 값을 만난다면 그 컬러 변환 시간을 크게 감소시킬 수가 있다.
* Open source 라이브러리인 OpenCV에 컬러 변환을 수행하는 함수들이 지원된다. 이러한 함수를 사용하여 위에서 실습한 컬러 변환을 해보고, 어떤 경우에 OpenCV와 같은 툴의 사용이 제한될 수 있는지에 관하여 기술한다. 제공되는 OpenCVcolorConversion 프로젝트를 이용한다.
* OpenCV에서는 컬러 변환을 수행하는 함수들이 지원된다. 이 때, 여러 용이한 함수를 이용해 이미지에 각종 처리를 할 수 있는데, 이 경우에 해당 함수 또는 라이브러리를 숙지하고 있다면 사용이 쉽겠으나, 그렇지 않다면 접근이 어렵다는 단점이 있다. 또한, opencv 자체의 환경이나 기존에 사용하던 함수 간의 호환성 역시 생각해야 하기 때문에, 이 역시 생각해야 한다.