대응점 및 히스토그램을 이용한 영상 간의 컬러 차이 측정 기법

황 영 배^{a)*}, 김 제 우^{a)}, 최 병 호^{a)}

Method of Measuring Color Difference Between Images using Corresponding Points and Histograms

Youngbae Hwang^{a)*}, Je Woo Kim^{a)}, and Byeong Ho Choi^{a)}

요 약

두 카메라 혹은 다수의 카메라에서의 컬러 보정은 이후 알고리즘의 성능 향상 및 양안식 3D 카메라에서 매우 중요한 기술이다. 최근 컬러 보정 방법들이 다수 제안되었지만 이 방법들의 결과에 대한 정확한 측정 방법이 많지 않으며 기존의 측정 방법은 두 영상이 카메라의 위치에 따른 서로 다른 장면을 가지고 있을 경우 적합하지 않을 수 있다. 본 논문에서는 컬러 보정을 위한 컬러 간의 차이 측정 기법을 제안한다. 이 기법은 대상이 되는 두 영상의 장면이 일치하지 않는 경우를 고려하여 대응점 검색을 통해 두 장면간의 같은 컬러를 가져야 하는 대응점을 찾고 이 대응점 주위의 영역으로부터 통계치를 계산하여 컬러의 차이를 계산한다. 이 경우 두 영상의 위치 변화를 하나의 기하학적 변환으로 설명하는 기존 방법에서 생길 수 있는 대응점간의 불일치를 고려할 수 있다. 또한 대응점들이 영상의 모든 영역을 포함하지 않을 수 있기 때문에 전체 영상의 통계치를 계산하여 컬러의 차이를 측정한다. 최종적인 컬러의 차이는 대응점 기반과 전체 영상 기반의 컬러 차이의 가중치의 합으로 결정되며 이 가중치는 대응점 기반의 컬러 비교가 영상 내의 얼마만큼의 영역을 포함하는지에 따라서 결정된다.

Abstract

Color correction between two or multiple images is very crucial for the development of subsequent algorithms and stereoscopic 3D camera system. Even though various color correction methods are proposed recently, there are few methods for measuring the performance of these methods. In addition, when two images have view variation by camera positions, previous methods for the performance measurement may not be appropriate. In this paper, we propose a method of measuring color difference between corresponding images for color correction. This method finds matching points that have the same colors between two scenes to consider the view variation by correspondence searches. Then, we calculate statistics from neighbor regions of these matching points to measure color difference. From this approach, we can consider misalignment of corresponding points contrary to conventional geometric transformation by a single homography. To handle the case that matching points cannot cover the whole regions, we calculate statistics of color difference from the whole image regions. Finally, the color difference is computed by the weighted summation between correspondence based and the whole region based approaches. This weight is determined by calculating the ratio of occupying regions by correspondence based color comparison.

Keyword : 컬러 차이(Color difference), 컬러 보정(Color correction), 양안식 3D 카메라(Stereoscopic 3D camera), 양안정합(Stereo matching)

a) 전자부품연구원

Korea Electronics Technology Institute(KETI)

* 교신저자 : 황영배 (ybhwang@keti.re.kr)

※ 본 연구는 지식경제부 및 한국산업기술평가관리원의 산업원천기술개발사업(산업기술)[10040246, 이동 로봇의 안정적 영상 획득을 통한 3D Depth 정보 획득과 실시간 객체 인식을 위한 로봇 비전 SoC 및 모듈 개발]의 일환으로 수행하였음.

· 접수일(2011년9월7일), 수정일(1차:2011년12월21일, 2차:2012년1월30일), 게재확정일(2012년2월2일)

I. 서론

두 카메라 혹은 다수의 카메라에서의 컬러 보정은 3D 복원이나 스테레오 매칭과 같은 컴퓨터 비전 알고리즘의 성능 향상을 위해서 필수적이고^[1] 양안식 3D 카메라에서 컬러 차이에 의한 이질감 및 피로도를 제거하기 위해서도 중요한 기술이다. 최근 다양한 컬러 보정 방법들이 제안되었고 이 방법들은 다양한 어플리케이션에 적용되어 좋은 결과를 보여주고 있다^[1-3]. 하지만 각 방법에서 사용되는 컬러 보정의 결과 비교를 위해 사용되는 측정 방법이 다르기 때문에 서로 간의 비교에 어려움을 겪고 있다. 컬러 패턴을 이용해서 다수의 카메라 간의 대응되는 컬러 샘플을 얻은 후 행렬 계산을 통해서 변환을 하는 방법^[1]에서는 샘플간 표준편차의 평균을 이용해서 컬러 보정의 결과를 판단하였다. 장면 보간(view interpolation)과 컬러 보정을 이용해서 다중 영상의 비디오 압축을 하는 연구^[2]에서는 PSNR(peak signal-to-noise ratio)을 이용하여 보간으로 얻어진 영상과 실제 영상과의 차이를 측정하였다. Tehrani et al.은 대응하는 특징점을 이용하여 다중 카메라의 컬러 보정을 반복적으로 수행하였는데 최종적인 컬러 보정의 결과를 측정하기 위해서 색상(hue)의 히스토그램의 유사성을 이용하였다^[3].

컬러 보정의 결과를 비교하기 위해서는 두 영상의 컬러의 차이를 측정해야 한다. 정확한 컬러의 차이 측정을 위해서는 두 영상 간의 실제로 대응하는 픽셀의 컬러를 알아야 하지만 카메라를 고정하고 얻은 영상이 아닌 장면 변화가 있을 때의 두 영상 간의 대응하는 픽셀을 구하는 것은 일반적인 스테레오 매칭 문제와 동일하게 된다. 스테레오 매칭은 복잡한 텍스처나 균일한 컬러, 반복적인 패턴을 가지고 있는 일반적인 영상 셋에 대해서 모든 픽셀의 대응점을 정확히 구하기 힘들다. 정확히 대응되지 않는 대응점을 사용하면 컬러의 차이에 대한 측정이 부정확하게 된다.

본 논문에서는 컬러 보정이 된 두 영상 간의 장면 차이가 있을 경우 실제 대응이 되는 영상 내 부분에 대해서 비교를 하기 위해서 특징점¹⁾(feature point)을 추출한 후 기술자(descriptor)를 이용하여 매칭을 수행한다. 그 후에 아웃라

이어(outlier)를 제거하기 위해서 호모그래피(homography)와 RANSAC(Random Sample Consensus)을 이용하여 가장 적합한 대응점을 결정한다. 기존에는 두 영상 간의 대응을 위해서 단일의 호모그래피만을 계산하여 각 픽셀 간의 대응점을 추출하였지만 장면과 카메라의 거리가 짧은 경우에는 이러한 단일 호모그래피만으로는 모든 픽셀 간의 대응점을 고려할 수 없다. 그러므로 본 논문에서는 불변 특징량을 추출하고 기술자를 이용한 매칭으로써 얻어진 대응점이 호모그래피로 계산된 대응점보다 정확하게 두 픽셀 간의 대응을 설명할 수 있다고 생각하였으며 기술자 간의 유사성으로 인해 생길 수 있는 아웃라이어는 호모그래피로 제거하도록 하였다. 이렇게 구해진 대응점 간의 비교는 대응점 주변 블록 간의 비교를 통해서 수행하였고 이 때 블록이 영상의 모든 영역을 포함하지 않을 수 있기 때문에 전체 영역에 대한 비교도 동시에 수행하였다. 최종적인 컬러 차이는 두 차이의 가중치의 합으로 계산되며 이 가중치는 블록이 전체 영역 대비 얼마만큼의 영역을 차지하는 지로 결정된다. 두 영상 간의 컬러 차이는 대응되는 픽셀 컬러 간의 RGB 유클리드 차이에 대한 평균 계산과 전체적인 영상 컬러 분포의 차이를 계산하는 히스토그램 비교를 수행한 후 이에 대한 합산치를 계산하여 픽셀 간의 직접적인 차이와 전체적인 분포의 차이를 모두 고려하였다.

II. 컬러 유사성 측정

컬러 유사성 측정은 그림 1과 같이 크게 대응점 검색 단계와 컬러 차이 측정 단계로 나뉜다. 대응점 검색 단계는 먼저 코너 부근과 컬러 블롭(color blob)안에서의 특징량 추출(feature detection), 특징량 매칭(feature matching) 그리고 잘못된 매칭(outlier) 제거 단계로 이루어져 있으며 컬러 유사성 측정 단계는 특징점 부근에서 통계치를 계산하고 두 영상 간의 겹치는 영역에서의 통계치를 계산한 후에 두 통계치의 가중치의 합으로 계산된다. 이 때 가중치는 특징점의 분포가 얼마나 영상의 영역을 포함하는지에 따라서 결정된다.

1) 본 논문에서는 영상에서 추출되는 feature를 특징량이라고 표현하며 이 특징량이 추출된 점을 특징점(feature point)라고 표시한다.

그림 1. 컬러 유사성 측정을 위한 알고리즘
Fig. 1. Proposed algorithm for color similarity measurement

특징량은 코너 영역에서 고유분석(eigen analysis)을 통해 추출되는 Harris 코너 [4]와 같은 코너점이나 코너 영역의 부분에서 추출되는 헤시안(Hessian) 및 라플라시안(Laplacian) 기반의 불변 특징량(SIFT^[5], SURF^[6])들이 주로 사용된다. 본 논문에서는 코너 부근의 불변 특징량으로 SIFT를 사용하였다. 하지만 이러한 특징량들은 코너 부근에서만 검출되기 때문에 영상 내의 다양한 컬러를 포함할 수 없을 수 있으며 코너점은 컬러의 변환점(transition)에 놓이는 경우가 많기 때문에 장면 컬러를 제대로 반영하지 못할 수도 있다. 영상이 균일한(homogeneous) 컬러를 갖는 영역이 많을 경우 영상 내의 주된 컬러에 대한 대응을 확보할 수 없다.

이러한 한계를 극복하기 위해서 코너 부근의 특징량 검출뿐만이 아니라 컬러 블롭에서의 특징량 검출도 동시에 수행하였다. 여기서 컬러 블롭이란 영상 내에서 비슷한 컬러

그림 2 특징량 추출 결과
Fig. 2. Results of feature extraction

러를 갖고 있는 영역을 의미하며 컬러 분할(color segmentation)과 같이 컬러가 유사한 영역을 찾아낸 후 그 컬러 블록에서 특징량을 검출하는 MSER(Maximally Stable Extremal Region)^[7]을 추가적으로 사용하였다. MSER은 영상의 문턱치(threshold)를 변화시켜가면서 얻어지는 이진화 영상(binary image)에서 발생하는 블록을 극한 영역(extremal region)이라고 할 때 그 영역을 포함하는 영역 중에서 가장 큰 영역을 최대 안정 극한 영역이라고 언급하며 특징량으로 검출하게 된다. 이렇게 추출된 MSER 영역에서 실제 특징점의 위치는 컬러 블록을 타원 적합(ellipse fitting)한 후의 원의 중심점이 된다.

그림 2에서는 입력 영상에 따른 특징량 추출 결과를 보여주고 있다. 그림 2(a)에서의 입력 영상에 대해서 그림 2(b)는 코너 부근에서 추출되는 SIFT^[5] 결과를 보여주고 있고, 그림 2(c)는 컬러 블록에서 추출되는 MSER^[6] 결과를 보여주고 있다. 그림에서 확인할 수 있듯이 SIFT의 경우에는 코너 부근에서 주로 추출되지만 컬러가 균일한 부분에서는 검출이 되지 않는 것을 확인할 수 있으며 MSER의 경우에는 강의 안쪽, 나무 사이의 빈 공간 그리고 바위의 가운데와 같은 주로 컬러가 균일한 영역에서 검출되는 것을 확인할 수 있다. 두 특징량이 서로 상호보완적인 관계를 가지고 있기 때문에 두 가지 특징량을 동시에 이용함으로써 영상 내의 다양한 컬러의 분포를 포함할 수 있도록 한다.

특징량의 검출 결과를 좀 더 자세히 보기 위해서 그림 3에서는 균일한 배경에서 촬영된 손바닥 영상에 대한 특징량 검출 결과를 보여준다. 그림 3(a)에서와 같이 특징량 추출에 많이 사용되는 Harris 코너의 경우에는 코너점이나 경

계선에서 주로 검출되는 것을 확인할 수 있다. 그림 3(b)에서의 SIFT 검출기는 Harris 코너에 비해서는 코너 안쪽이나 경계선 안쪽에서 검출이 되는데 그 이유는 DoG (Difference of Gaussian)가 스케일 공간(scale space)에서 두 개의 Gaussian의 차이를 최대화 혹은 최소화하는 곳에서 추출되기 때문에 실제로 스케일 공간에서는 코너점이나 경계선에서 추출되지만 원래의 해상도(resolution)에서 그 위치를 표시하면 코너점이나 경계선에서 어느 정도 멀어지게 된다. 그림 3(b)에서 원의 크기는 그 특징점의 스케일을 의미하며 스케일이 커질수록 낮은 해상도에서 검출된 것이다. 그림 3(c)에서 MSER 추출 결과는 MSER이 균일한 영역의 중심에서 주로 검출이 된다는 것을 확인할 수 있다. 손가락의 마디나 손바닥에서 주로 추출되는 것을 볼 수 있으며 SIFT와는 위치가 서로 다른 것을 확인할 수 있다. 또한 손의 끝에서 위치가 비슷한 특징량의 경우에도 SIFT는 큰 스케일을 갖기 때문에 더 많은 영역을 포함하지만 MSER의 경우에는 SIFT에 비해서 더 작은 스케일을 갖는 것을 확인할 수 있다. MSER에서 컬러 블록을 계산할 때 서로 같은 영역을 결정하는 기준의 파라미터, Δ 가 존재하며 일반적으로 사용되는 값은 5이다.

특징량을 추출한 후에는 특징량 간의 매칭을 수행하게 된다. 영상 간의 변화가 작다면 템플릿 매칭(template matching)으로도 좋은 결과를 얻을 수 있지만 영상 간의 움직임이 큰 경우에도 정확한 매칭 결과를 얻기 위해서는 기술자(descriptor) 기반의 매칭을 수행해야 한다. 또한 MSER이 균일한 영역에서 얻어지기 때문에 템플릿 매칭의 블록 간의 비교로는 정확한 대응점을 찾기 힘들다는 단점이 있

그림 3. 균일한 물체에 대한 특징량 추출 결과

Fig. 3. Results of feature extraction for a homogeneous object)

다. 특징점 근처를 기술자로 기술한 후에 두 영상 간의 가장 비슷한 기술자를 갖는 경우에 두 특징점을 매칭된 점으로 결정하는 방법이다.

여기서는 SIFT 기술자를 사용하는데 SIFT 기술자는 영상에서의 도함수(gradient)를 구한 후 영역을 나눠서 영역별로 방위의 히스토그램을 구하고 그 히스토그램들을 모두 합쳐서 전체 영역을 기술하게 된다. 이러한 방위의 히스토그램은 영상의 조명 변화나 밝기 변화에 불변한 특징을 가지고 있으며 영상이 회전을 하였다 하더라도 전체 방위를 계산한 후에 이를 통해 보정을 하면 회전에 대해서도 불변한 특성을 유지할 수 있다. 그림 4에서는 SIFT를 이용하여 각 특징점을 기술한 후에 매칭한 결과를 보여주고 있다. 일반적으로 템플릿 매칭에 사용되는 검색 범위의 제한 없이도 기술자를 통해 대부분의 특징량이 정확하게 매칭이 되는 것을 확인할 수 있지만 두 특징점 간의 대각선 매칭이 존재하며 이는 아웃라이어(outlier)이기 때문에 제거하지 않는다면 후의 컬러 유사성 측정에서 큰 오차를 범할 수 있게 된다.

특징량 매치에서 생길 수 있는 잘못된 매칭을 제거하기 위해서는 아웃라이어 제거를 수행해야 한다. 아웃라이어 제거를 위해서는 영상 내의 두 평면 사이의 관계를 설명하는 기하학적인 모델인 호모그래피(homography)와 확률적 투표 기반의 RANSAC을 이용하였다. 그림 5에서는 그림 4에서의 기술자 기반 매칭 결과에 호모그래피에 기반한 RANSAC을 적용한 결과를 보여주고 있다. 그림 4와 비교해서 아웃라이어들이 모두 제거된 것을 확인할 수 있다. 일반적으로 장면과 카메라가 가까운 경우에 단일의 호모그래피로 두 영상 간의 대응점을 제대로 설명할 수 없을 수 있다. 기존의 논문에서는 먼저 두 영상 간의 단일 호모그래피를 구한 후에 한 영상의 특징점을 다른 영상의 특징점으로 변환하여 두 영상 간의 대응점을 계산하였다. 하지만 본 논문에서는 먼저 특징점들간의 대응 관계를 구한 후에 호모그래피를 구해서 아웃라이어 제거만을 하였다. 그러므로 그림 4에서와 같이 일반적으로는 매칭이 잘 이루어지고 아웃라이어의 경우에는 큰 기하학적 에러를 갖는다면 아웃라이어 제거를 통해서 장면과 카메라 간의 거리가 가까운 영상이라 하더라도 얻어진 대응점은 정확할 수 있다. 이러한 대

응점이 영상 전체를 포함할 수는 없기 때문에 대응점 기반으로 좀 더 정확한 컬러 차이를 측정하고자 한다면 단일 호모그래피가 아닌 좀 더 일반적인 기하학적인 모델을 이용하여 대응점을 구하는 것이 낫다.

그림 4. SIFT 기술자 기반 특징량 매칭 결과
Fig. 4. Results of SIFT descriptor based feature matching

그림 5. Outlier 제거 결과
Fig. 5. Results of outlier rejection

대응점 검색 단계가 끝나면 컬러 차이 측정(color difference measurement) 단계가 진행된다. 컬러 유사성 측정은 크게 대응점 기반과 전체 영상 기반으로 나눌 수 있다. 대응점 기반은 특징점이 매칭되었을 때 그 특징점을 포함하는 NxN의 블록을 만들어서 그 블록 간의 통계치를 비교하는 것이다. 블록의 크기 N은 1인 경우에는 특징점에 대한 비교가 이루어지게 되고 N이 영상 사이즈만큼 클 경우에는

그림 6. 컬러 유사성 비교를 위해 서로 비교되는 블록들(같은 색일 경우 대응되는 블록임)
Fig. 6. Corresponding blocks for color similarity measurement

영상 전체 영역에 대한 비교가 된다. 영상의 사이즈가 커지더라도 그 크기에 따라서 특징점의 수가 많아지기 때문에 블록의 크기 N 은 달라질 필요가 없으며 N 은 대략적으로 특징점 근처의 통계를 표현하기에 적합하면 되기 때문에 그 값의 변화에 따라서 크게 민감하지 않다. 그림 6에서는 50×50 의 블록을 만들었을 경우를 보여주고 있으며 같은 색의 사각형끼리 서로의 통계치를 계산해서 비교하게 되며 본 논문에서의 N 은 모두 50을 사용하였다.

컬러 차이를 계산하기 위해 이용하는 통계치는 영역 내에서 픽셀 간의 유클리드 거리의 평균과 히스토그램(histogram) 비교이다. 유클리드 거리의 평균에 의한 비교는 (1)과 같다. (1)에서 I_{ik}, I_{jk} 는 각각 영상 i 와 영상 j 의 k 번째 대응 컬러를 의미하고 실제 두 컬러 간의 차이는 (1)의 아래 식에서처럼 RGB 각 채널의 유클리드 거리를 계산한 후 최대값인 255로 정규화(normalized)한다. 블록 간의 유클리드 거리는 블록에 대해서 평균을 내기 위해서 모두 합한 후에 블록의 픽셀 수로 나누어지며 전체 블록에 대한 컬러 차이는 이렇게 구해진 유클리드 거리의 평균을 다시 계산한다.

$$d_{color}(i, j) = \frac{\sum_{k=1}^N \left[\sum_{block} \| I_{ik} - I_{jk} \| \right] / N_{block}}{N} \quad (1)$$

$$\| I_{ik} - I_{jk} \| = \sqrt{(I_{ik}^R - I_{jk}^R)^2 + (I_{ik}^G - I_{jk}^G)^2 + (I_{ik}^B - I_{jk}^B)^2} / 255$$

히스토그램의 비교는 먼저 각 블록에서 히스토그램을 구한 후에 두 히스토그램을 비교함으로써 두 블록의 차이를 측정하게 된다. 히스토그램을 비교하는 방법은 다양하게 존재하는데 각 bin간의 직접적인 차이를 이용하는 방법, 비계수적인(non-parametric) 비교 방법, 정보이론(information theory)을 이용한 비교방법, ground 거리 측정이 사용된다. 본 논문에서는 비계수적인 비교 방법 중의 하나인 χ^2 (chi-square) 거리를 이용하며 계산은 (2)와 같다^[8].

$$d(H_i, H_j) = \frac{1}{2} \sum_{l=1}^B \frac{|h_i(l) - h_j(l)|^2}{h_i(l) + h_j(l)} \quad (2)$$

(2)에서 H_i, H_j 는 대응되는 두 영역의 히스토그램을 의미하고 B 는 bin의 수를 의미한다. $h_i(l)$ 은 l 번째 bin에서의 히스토그램의 값을 의미한다.

대응되는 두 영역 간의 컬러의 차이의 측정은 (3)와 같이 유클리드 거리의 평균과 히스토그램의 차이의 가중치의 합으로 나타낸다. 히스토그램의 차이도 (1)의 유클리드 거리의 평균과 마찬가지로 N 개의 대응하는 블록의 수로 나누어서 블록 간의 평균을 계산하고 채널 간의 관계도 더한 후에 채널 수로 나누어진다. (3)에서 가중치 α_1 는 유클리드 거리의 차이와 히스토그램 차이의 비중을 나타내며 유클리드 거리 및 히스토그램 모두 정규화를 하였기 때문에 가중치는 픽셀 간의 직접적인 비교인 유클리드 거리와 전체적인 분포 비교인 히스토그램 비교 중 어느 통계치를 더 중요하게 생각할 것인지에 대한 값이며, 본 논문에서는 $\alpha_1 = 0.1$ 을 이용한다.

$$d_{block}(i, j) = d_{color}(i, j) + \alpha_1 \cdot \left\{ \sum_{c=R, G, B} \left[\sum_{k=1}^N d_k^c(H_i, H_j) \right] / N \right\} / 3 \quad (3)$$

블록 간의 컬러 비교는 그림 6에서 볼 수 있듯이 영상 내의 다양한 컬러 분포를 포함하지 못할 수 있다. 그렇기 때문에 전체 영역에 대한 컬러 비교를 수행하여야 한다. 하지만 두 영상의 겹치는 부분이 영상 내의 일부분일 수 있기 때문에 두 영상의 겹치는 영역(overlapped region)을 계산하여 그 부분에 대해서만 컬러 차이를 측정해야 한다. 겹치는 영역 계산은 앞의 아웃라이어 제거에서 구했던 호모그래피를 이용한다. 영상 i 의 모든 점을 호모그래피를 이용해서 영상 j 로 옮길 수 있고 이 때 영상 j 안에 존재하는 점들이 겹치는 영역이 된다. 마찬가지로 영상 j 에서의 모든 점을 영상 i 로 옮길 수 있으며 이 때 영상 i 에 존재하는 점들이 겹치는 영역이 된다. 각 영상에서의 겹치는 영역을 찾았으면 블록 간의 컬러 비교와 마찬가지로 (4)와 같이 픽셀 간의 유클리드 거리의 평균 및 히스토그램의 차이를 계산하여 전체 영상에 대한 컬러 차이 측정을 수행한다. 가중치 α_2 는 α_1 와 마찬가지로 0.1를 이용한다.

$$d_{image}(i, j) = \left\{ \sum_{\text{Overlaped Region}} \|I_i - I_j\| \right\} / N_{\text{Overlap}} + \alpha_2 \cdot \left\{ \sum_{c=R, G, B} d^c(H_i, H_j) \right\} / 3 \quad (4)$$

전체 영상에 대한 컬러 차이 측정은 블록 기반의 측정에 비해서 사용하는 픽셀의 수가 많고 정확하게 대응되는 것이 아니기 때문에 블록 기반에 비해서 통계치 자체의 정확성은 부족할 수 있다. 하지만 블록 기반이 영상 내의 모든 영역을 포함하고 있는 것이 아니므로 영상의 전체적인 비교를 위해서는 필요하다. 본 논문의 두 영상 간의 컬러 차이는 (5)와 같이 전체 영상에 대한 컬러 차이와 블록 기반의 컬러 차이 측정의 가중치의 합으로 이루어진다. 이 때 두 컬러 차이의 중요도의 비를 나타내는 β 는 측정에 의해서 결정되게 된다. 이 측정은 기본적으로 0.1이라는 최소값을 갖고 영상 i 및 j 에서의 각 블록의 합집합의 영역이 전체 겹치는 영역에 비해서 얼마의 비율을 가지고 있는지에 대한 값으로 결정된다. 블록의 합집합의 영역이 전체 영역을 모두 포함했을 경우에 최소값인 0.1의 값을 갖게 되고 반대로 블록이 하나도 존재하지 않을 경우에는 최대값인 0.9의 값을 갖게 된다. 일반적으로는 각 블록의 합집합의 영역이 절반 이상은 될 것이기 때문에 0.5 이하의 값을 가지게 될 것이다. 컬러 차이를 측정해야 하는 영상 셋이나 실험 환경에 따라서 파라미터들은 변경될 수 있다.

$$d_{total}(i, j) = d_{block}(i, j) + \beta \cdot d_{image}(i, j) \quad (5)$$

(6)에서 $OverlapRegion_i$ 는 영상 i 에서의 겹치는 영역을 의미하고 $Region_i(k)$ 는 영상 i 에서 k 번째 블록의 영역을 의미한다. 각 영역의 합집합의 영역을 겹치는 영역으로 나누면 정규화된 블록의 영역을 알 수 있으며 이 값을 1에서 빼서 블록의 영역이 전체 합집합의 영역에서 포함하지 못하는 영역을 계산하였다. 두 영상 간의 겹치는 영역이 서로 다를 수 있기 때문에 각 영상에 대해서 이 포함하지 못하

는 영역을 계산하였고 이 β 의 값이 크다는 것은 블록이 포함하는 영역이 작다는 것을 의미하기 때문에 전체적인 컬러 비교를 하는 $d_{image}(i, j)$ 의 값의 비중을 더 크게 한다.

III. 실험 결과

2장에서 컬러 차이 측정은 영상이 R, G, B의 값을 갖는 경우에 대해서만 설명하고 있다. 영상이 그레이 스케일 (grey-scale)일 경우에는 c 가 한 개인 경우이므로 (3)과 (4)에서 유클리드 거리 대신에 그레이 스케일 간의 절대값을 계산하게 되고 히스토그램 차이의 경우에는 채널 간의 합산 대신에 하나의 채널에 대해서만 적용하면 된다. 컬러 차이를 측정하기 위해서 그림 7과 같이 비슷한 장면을 가지면서 서로 컬러가 다른 영상 셋을 이용하였다. 첫 번째 그림과 두 번째 그림의 경우에는 방송용 스테레오 카메라 시스템으로 획득된 영상이고 세 번째 그림은 PointGrey Grasshopper2와 Olympus Pen3로 비슷한 위치에서 촬영된 영상이다. 컬러 보정 알고리즘은 다양한 특징량과 기술자를 통해 대응점을 추출한 후에 대응점 간의 선형 혹은 비선형 변환을 계산한 알고리즘을 사용하였는데^[9], 논문에서 가장 좋은 결과를 보여주는 SIFT 특징량 및 SIFT 기술자를 통한 매칭, 그리고 일반적 다항식을 이용한 컬러 보정에 의한 방법을 적용하였다.

그림 8에서는 그림 7의 입력 영상에 대한 컬러 보정 결과를 보여주고 있다. 그림 7에서 사람의 시각으로 확인할 수 있는 컬러 간의 차이를 그림 8에서 어느 정도 제거하는 것을 확인할 수 있었다. 하지만 호모그래피로 표현될 수 있는 정면의 장면에 비해 주변 배경 부분은 컬러 차이가 더 나는 것을 볼 수 있는데 이것은 단일 호모그래피 방식의 컬러 보정의 한계를 보여준다. 본 논문에서도 비슷한 방식의 단일 호모그래피 방식을 이용하여 영상 간의 대응점을 구하게 되는데 이러한 대응점은 호모그래피로 비교적 정확하

$$\beta = 0.1 + 0.4 \times \left(1 - \frac{\bigcup_{k=1}^N Region_i(k)}{OverlapRegion_i} \right) + 0.4 \times \left(1 - \frac{\bigcup_{k=1}^N Region_j(k)}{OverlapRegion_j} \right) \quad (6)$$

그림 7. 컬러 차이 측정을 위한 테스트 영상 셋
Fig. 7. Test image sets for measruing color difference

그림 8. 컬러 보정한 결과
Fig. 8. Result of Color Correction

게 설명할 수 있는 정면 부분에 많이 놓이게 될 것이다. 이러한 한계를 극복하기 위해서 본 논문에서는 히스토그램을 이용한 전체적인 컬러 차이를 측정하였다. 컬러 보정 전후의 컬러 차이를 정량적으로 판단할 수 있는지 확인하기 위해서 제안한 컬러 차이 측정 방법을 적용하였다. 사람 시각에서 분명 그림 7보다 그림 8의 대응되는 영상의 컬러가 더 비슷하기 때문에 측정 결과 상 컬러 거리가 줄어들어야 한다.

표 1에서는 입력 영상에 대한 컬러 차이 측정 결과를 보여주고 있고 표 2에서는 컬러 보정된 영상에 대한 컬러 차이 측정 결과를 보여주고 있다. 비교를 위해서 기존에 많이 사용되는 PSNR을 이용한 비교^[2]와 색상 히스토그램 비교^[3]의 결과도 함께 계산하였다. Yamamoto et al. [2]에서는 동일한 영상의 비교를 수행했기 때문에 PSNR을 직접적으로 계산할 수 있었지만 본 논문에서는 영상 간의 장면변화가 있기 때문에 두 영상간의 관계를 나타내기 위해서 호모그래피를 통해서 대응점을 구하였다. 표 1과 2에서는 최종적인 d_{total} 을 계산하기 위해서 계산되는 d_{block} , d_{image} 및 β 값을 표시하였고 d_{block} 과 d_{image} 를 계산하기 위해서 내부적으로 계산되는 컬러 차이 d_{color} 와 히스토그램 차이 d_{hist} 도

계산하였다.

PSNR의 경우에는 높은 dB값을 갖는 것이 더 작은 차이를 나타내므로 영상 셋 모두에 대해서 컬러 거리가 줄어드는 결과를 보였다. 하지만 색상 히스토그램의 경우에는 영상 셋 1에 대해서는 그 차이가 줄어들었지만 영상 셋 2와 3에 대해서는 그 차이가 오히려 커지는 것을 확인할 수 있었다. 그러므로 색상 히스토그램은 컬러 차이를 나타내는데 적합하지 않음을 알 수 있었다. 제안한 방법도 PSNR과 마찬가지로 더 작은 차이를 나타내는 것을 확인할 수 있었다. 제안한 방법의 실제적인 의미는 블록 간의 비교를 통해서 지역적(local)인 비교와 전체 영역 간의 비교를 통한 전역적(global)인 비교를 동시에 수행하고 이에 대한 가중치를 두어서 선택적으로 그 차이를 결정하는 것이다. 실제로 영상 셋 3의 경우에 카메라로부터 상당히 근사한 거리에서 촬영된 영상이기 때문에 호모그래피로 설명할 수 없는 영역들이 많이 존재하게 된다. 그러므로 전체적인 영역 비교의 가중치가 높아져야 하고 실제 실험 결과에서도 β 값이 커지는 것을 확인할 수 있다. 또한 PSNR 계산 방법은 픽셀과 픽셀 간의 비교, 색상 히스토그램은 전체적인 비교만을 하는데 반해서 제안한 방법은 두 통계치를 모두 사용함으

표 1. 입력 영상에 대한 컬러 차이 측정 결과

Table 1. Result of Measuring Color Difference for Input Images

	PSNR[2]	Hue Histogram[3]	d_{block}		d_{image}		β	d_{total}
			d_{color}	d_{hist}	d_{color}	d_{hist}		
영상 셋 1	24.02dB	0.240	0.211		0.150		0.197	0.142
			0.088	0.249	0.081	0.688		
영상 셋 2	17.90dB	0.276	0.104		0.340		0.410	0.244
			0.093	0.118	0.129	2.106		
영상 셋 3	7.69dB	0.214	0.545		1.217		0.578	1.249
			0.303	2.422	0.412	8.046		

표 2. 컬러 보정된 영상에 대한 컬러 차이 측정 결과

Table 2. Result of Measuring Color Difference for Color Corrected Images

	PSNR[2]	Hue Histogram[3]	d_{block}		d_{image}		β	d_{total}
			d_{color}	d_{hist}	d_{color}	d_{hist}		
영상 셋 1	26.68dB	0.173	0.111		0.071		0.206	0.125
			0.085	0.254	0.050	0.210		
영상 셋 2	22.57dB	0.401	0.096		0.100		0.443	0.141
			0.089	0.074	0.083	0.172		
영상 셋 3	12.13dB	0.328	0.632		0.718		0.561	1.035
			0.419	2.130	0.251	4.676		

로써 한 가지 방법만으로는 표현할 수 없는 차이를 설명할 수 있다.

IV. 결 론

본 논문에서는 두 영상 간의 컬러 차이 측정을 위한 방법을 제안하였다. 영상 간의 장면이 다를 수 있기 때문에 이를 고려하기 위해서 특징량 기반의 매칭 방법을 사용하였고, 다양한 컬러 분포를 포함하기 위해서 코너 부근의 특징량과 컬러 블롭 기반의 특징량을 모두 사용하였다. 컬러 차이 계산에서는 각 매칭점을 중심으로 한 블롭 기반의 통계치 비교와 전체 영상에서의 통계치 비교를 하였으며 두 계산값 사이의 가중치는 블롭들이 전체 영상 영역 중에 얼마만큼을 차지하는 지에 대한 것으로 결정하였다. 통계치로는 픽셀 간의 유클리드 거리의 평균과 히스토그램 차이를 사용하였다. 제안한 컬러 차이 측정 방법은 카메라와 카메라 간의 컬러 차이를 측정해야 하는 어플리케이션에 사용될 수 있으며 특히 컬러 보정 전후의 컬러 차이 비교에 적용가능함을 확인하였다. 또한 지역적인 통계치와 전역적인 통계치의 합산, 픽셀과 픽셀 간의 비교와 전체적인 히스토그램 비교를 통해서 영상 간의 컬러 비교를 종합적으로 할 수 있도록 제안되었다.

참 고 문 헌

- [1] A. Ilie, G. Welch, "Ensuring color consistency across multiple cameras," IEEE International Conference on Computer Vision, vol.2, pp.1268-1275, Oct. 2005
- [2] K. Yamamoto, M. Kitahara, H. Kimata, T. Yendo, T. Fujii, M. Tanimoto, S. Shimizu, K. Kamikura, Y. Yashima, "Multiview Video Coding Using View Interpolation and Color Correction," IEEE Transactions on Circuits and Systems for Video Technology, vol.17, no.11, pp.1436-1449, Nov. 2007
- [3] M. P. Tehrani, A. Ishikawa, S. Sakazawa, A. Koike, "Iterative colour correction of multicamera systems using corresponding feature points," Journal of Visual Communication and Image Representation, vol.21, no.5-6, pp.377-391, 2010
- [4] C. Harris and M. Stephens, "A combined corner and edge detector," Proceedings of the 4th Alvey Vision Conference, pp. 147 - 151, 1988
- [5] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints", International Journal of Computer Vision, vol. 60, no. 2, pp. 91-110, 2004.
- [6] Bay, H., Tuytelaars, T., and Gool, L.V., "SURF: Speeded Up Robust Features", Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346-359, 2008
- [7] J. Matas, O. Chum, M. Urba, and T. Pajdla. "Robust wide baseline stereo from maximally stable extremal regions." Proc. of British Machine Vision Conference, pp 384-396, 2002
- [8] Greenwood, P.E., Nikulin, M.S., A guide to chi-squared testing. Wiley, New York
- [9] Youngbae Hwang, Je Woo Kim, Byeong Ho Choi, and WangHeon Lee, "Color correction without color patterns for stereoscopic camera systems", International Conference on Control, Automation and System (ICCAS), 2011

저 자 소 개

황 영 배

- 2001년 : KAIST 전자전산학과 학사
- 2003년 : KAIST 전자전산학과 석사
- 2009년 : KAIST 전자전산학과 박사
- 2009년 ~ 2011년 : 삼성테크윈 로봇사업그룹 책임연구원
- 2011년 ~ 현재 : 전자부품연구원 멀티미디어IP센터 선임연구원
- 주관심분야 : 컬러 보정, 노이즈 모델링, 3DTV

저 자 소 개

김 제 우

1997년 : 서울시립대학교 제어계측공학과 학사
1999년 : 서울시립대학교 제어계측공학과 석사
1999년 ~ 현재 : 전자부품연구원 멀티미디어IP연구센터 책임연구원
주관심분야 : 멀티미디어 코덱, 비디오 코딩 알고리즘 및 최적화 구현

최 병 호

- 1991년 : 한양대학교 전자공학과 학사
- 1993년 : 한양대학교 전자공학과 석사
- 2005년 ~ 2010년 : 중앙대학교 첨단영상대학원 영상공학과 박사
- 1993년 ~ 1997년 : LG전자 주임연구원
- 1997년 ~ 현재 : 전자부품연구원 멀티미디어IP연구센터 센터장
- 2005년 ~ 2008년 : 3DTV분과위원, 3D비디오코덱WG 의장
- 2009년 ~ 현재 : 실감방송분과위원
- 2009년 ~ 현재 : 실감미디어산업협회 부회장
- 주관심분야 : 3DTV, UDTV, 방송장비