

컬러 보정을 위한 컬러 간의 유사성 측정 기법

황영배, 김제우, 최병호
전자부품연구원

ybhwang@keti.re.kr, jwkim@keti.re.kr, bhchoi@keti.re.kr

Method of Measuring Color Similarity for Color Correction

Youngbae Hwang, Je Woo Kim, Byeong Ho Choi
Korea Electronics Technology Institute (KETI)

요 약

두 카메라 혹은 다수의 카메라에서의 컬러 보정은 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 경우에 적합하지 않을 수 있다. 본 논문에서는 컬러 보정을 위한 컬러 간의 유사성 측정 기법을 제안한다. 이 기법은 대상이 되는 두 영상의 장면이 일치하지 않는 경우를 고려하여 대응점 검색을 통해 두 장면 간의 같은 컬러를 가져야 하는 대응점을 찾고 이 대응점 주위의 영역으로부터 통계치를 계산하여 컬러의 유사성을 비교한다. 이 경우 두 영상의 위치 변화에 따른 장면 변화와 대응점의 약간의 어긋남에 대해서 고려할 수 있다. 또한 대응점들이 영상의 모든 영역을 포함하지 않을 수 있기 때문에 전체 영상의 통계치를 계산하여 컬러의 유사성을 비교도 동시에 수행하여 결과적인 컬러의 유사성은 대응점 기반과 전체 영상 기반의 유사성의 가중치의 합으로 결정되며 이 가중치는 대응점 기반의 컬러 비교가 영상 내의 얼마만큼의 영역을 포함하는지에 따라서 결정된다.

1. 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. [3]에서는 대응하는 특징점을 이용하여 다중 카메라의 컬러 보정을 반복적으로 수행하였는데 최종적인 컬러 보정의 결과를 측정하기 위해서 히스토그램의 유사성을 이용하였다.

컬러 보정의 결과를 비교하기 위해서는 두 영상의 컬러의 유사성을 측정해야 한다. 정확한 컬러의 유사성 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 사용한다면 컬러의 유사성의 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 특징점을 추출한 후 호모그래피(homograph)와 RANSAC(Random Sample Consensus)를 이용하여 가장 적합한 대응점을 추출한다. 하지만 특징점을 뽑는 과정에서나 매칭을 하는 과정에서 한두 픽셀 정도의 어긋남이 발생할 수 있기 때문에 대응점의 두 픽셀 간의 비교를 직접 하지 않고 대응점을 포함하는 영역에서 평균값을 계산하여 비교한다. 그리고 그 영역에서의 히스토그램을 구해서 비교한다.

2. 컬러 유사성 측정

컬러 유사성 측정은 그림 1 과 같이 크게 대응점 검색 단계와 컬러 유사성 측정 단계로 나뉜다. 대응점 검색 단계는 먼저 코너 부근과 컬러 블롭(color blob)안에서의 특징량 추출, 특징량 매칭 그리고 잘못된 매칭(outlier) 제거 단계로 이루어져 있으며 컬러 유사성 측정 단계는 특징량 부근에서 통계치를 계산하고 두 영상 간의 겹치는 영역에서의 통계치를

계산한 후에 두 통계치의 가중치의 합으로 계산된다. 이 때 가중치는 특징량의 분포가 얼마나 영상의 영역을 포함하는지에 따라서 결정된다.

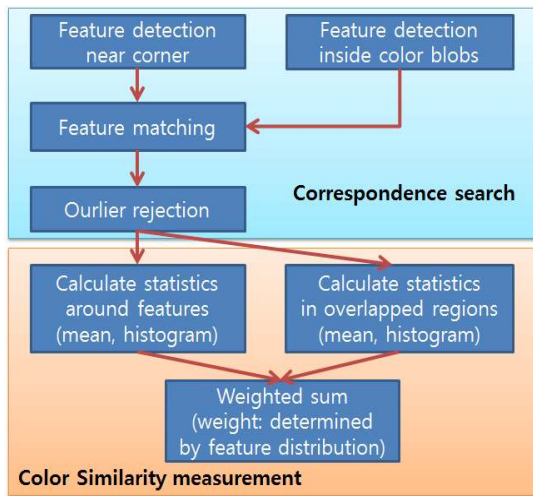


그림 1. 컬러 유사성 측정을 위한 알고리즘

특징량은 코너 영역에서 고유분석(eigen analysis)을 통해 추출되는 Harris 코너 [4]와 같은 코너점이나 코너 영역의 부분에서 추출되는 헤시안(Hessian)이나 라플라시안(Laplacian) 기반의 불변 특징량(SIFT [5], SURF [6])들이 주로 사용된다. 본 논문에서는 코너 부근의 불변 특징량으로 SIFT 를 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 검출뿐만 아니라 컬러 블롭에서의 특징량 검출도 동시에 수행하였다. 컬러 분할(color segmentation)과 같이 컬러가 유사한 영역을 찾아낸 후 그 컬러 블롭에서 특징량을 검출하는 MSER(Maximally Stable Extremal Region) [7]을 추가적으로 사용하였다. MSER 은 영상의 문턱치(threshold)를 변화시켜가면서 얻어지는 이진화 영상(binary image)에서 발생하는 블롭을 극한 영역(extremal region)이라고 할 때 그 영역을 포함하는 영역 중에서 가장 큰 영역을 최대 안정 극한 영역이라고 언급하며 특징량으로 검출하게 된다.

그림 2 에서의 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 블롭에서 추출되는 MSER[6] 결과를 보여주고 있다. 그림에서 확인할 수 있듯이 SIFT 의 경우에는 코너 부근에서 주로 추출되지만 컬러가 균일한 부분에서는 검출이 되지 않는 것을 확인할 수 있으며 MSER 의 경우에는 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.



(a) 입력 영상



(b) 코너 부근에서 추출되는 불변 특징량(SIFT) 결과



(c) 컬러 블롭에서 검출되는 불변 특징량(MSER) 결과

그림 2 특징량 추출 결과

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 MSER 이 균일한 영역에서 얻어지기 때문에 템플릿 매칭의 블롭 간의 비교로는 정확한 대응점을 찾기 힘들다는 단점이 있다. 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 3 에서의 SIFT 를 이용하여 각 특징점을 기술한 후에 매칭한 결과를 보여주고 있다. 일반적으로 템플릿 매칭에 사용되는 검색 범위의 제한이 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 RANSAC 을 이용하였다. 그림 4 에서는 그림 3 에서의 기술자 기반 매칭 결과에 호모그래피에 기반한 RANSAC 을 적용한 결과를 보여주고 있다. 그림 4 와 비교해서 아웃라이어들이 모두 제거된 것을 확인할 수 있다.

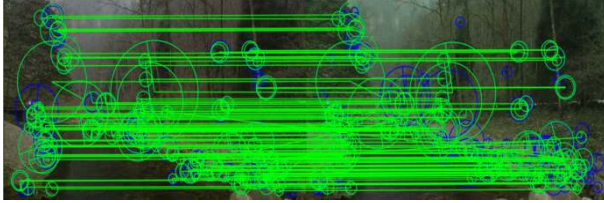


그림 3 SIFT 기술자 기반 특징량 매칭 결과

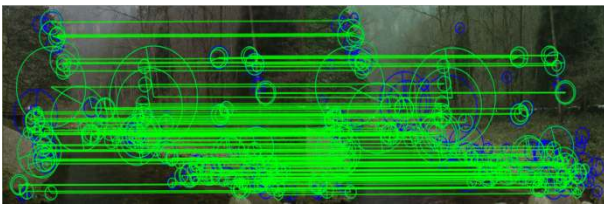


그림 4 Outlier 제거 결과

대응점 검색 단계가 끝나면 컬러 유사성 측정(color similarity measurement) 단계가 진행된다. 컬러 유사성 측정은 크게 대응점 기반과 전체 영상 기반으로 나뉠 수 있다. 대응점 기반은 특징점이 매칭되었을 때 그 특징점을 포함하는 NxN 의 블록을 만들어서 그 블록 간의 통계치를 비교하는 것이다. 그림 5 에서는 60x60 의 블록을 만들었을 경우를 보여주고 있으며 같은 색의 사각형끼리 서로의 통계치를 계산해서 비교하게 된다.

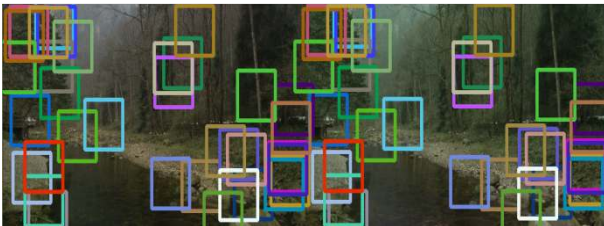


그림 5 컬러 유사성 비교를 위해 서로 비교되는 블록들 (같은 색일 경우 대응되는 블록임)

컬러 유사성 비교를 하기 위해 이용하는 통계치는 정규화된 평균(normalized mean)과 히스토그램(histogram) 비교이다. 정규화된 평균에 의한 비교는 수식 1 과 같다. 식 1 에서 μ_{ik}, μ_{jk} 는 각각 영상 i 와 영상 j 의 k 번째 대응 블록의 평균을 의미하며 σ_{ij}, σ_{jk} 는 각각 영상 i 와 영상 j 의 k 번째 대응 블록의 표준편차를 의미한다. 평균이 같더라도 컬러 분포가 동일하다고 볼 수 없기 때문에 표준편차로 나눠서 정규화된 평균을 사용하였다. 전체 블록에 대한 컬러 유사성은 식 1 에서처럼 정규화된 평균의 차이에 대한 L_2 norm 을 계산한다.

$$S_{\mu}(i, j) = \sqrt{\sum_{k=1}^N \left(\frac{\mu_{ik} - \mu_{jk}}{\sigma_{ik} \sigma_{jk}} \right)^2} \quad (\text{식 1})$$

히스토그램의 비교는 먼저 각 블록에서 히스토그램을 구한

후에 두 히스토그램을 비교함으로써 두 블록의 유사성을 측정하게 된다. 히스토그램을 비교하는 방법은 다양하게 존재하는데 각 bin 간의 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 식 2 와 같다.

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

식 2 에서 H_i, H_j 는 대응되는 두 영역의 히스토그램을 의미하고 B는 bin 의 수를 의미한다. $h_i(l)$ 은 l번째 bin 에서의 히스토그램의 값을 의미한다.

대응되는 두 영역 간의 컬러의 유사성의 측정은 식 3 와 같이 정규화된 평균의 차이와 히스토그램의 차이의 가중치의 합으로 나타내어 진다. 히스토그램의 차이도 식 1 의 정규화된 평균의 차이와 마찬가지로 N 개의 대응하는 블록의 수에 대해서 L_2 norm 의 형태를 갖는다. 식 3 에서 가중치 α_1 는 정규화된 평균의 차이와 히스토그램 차이의 비율을 나타내며 일반적인 경우는 $\alpha_1 = 0.5$ 를 이용한다.

$$S_{\text{block}}(i, j) = S_{\mu}(i, j) + \alpha_1 \cdot \sqrt{\sum_{k=1}^N [d_k(H_i, H_j)]^2} \quad (\text{식 3})$$

블록 간의 컬러 비교는 그림 5 에서 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 유사성을 측정해야 한다 겹치는 영역 계산은 앞의 아웃라이어 제거에서 구했던 homography 를 이용한다. 영상 i 의 모든 점을 homography 를 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 식 4 와 같이 정규화된 평균 및 히스토그램의 차이를 계산하여 전체 영상에 대한 유사성 측정을 수행한다.

$$S_{\text{image}}(i, j) = \left| \frac{\mu_i - \mu_j}{\sigma_i \sigma_j} \right| + \alpha_2 \cdot d(H_i, H_j) \quad (\text{식 4})$$

전체 영상에 대한 유사성 측정은 블록 기반의 유사성 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 유사성의 전체적인 비교를 위해서는 필요하다. 본 발명의 두 영상 간의 컬러 유사성은 식 5 와 같이 전체 영상에 대한 유사성과 블록 기반의 유사성 측정의 가중치의 합으로 이루어진다. 이 때 두 유사성의 중요도의 비를 나타내는 β 는 측정에 의해서 결정되게 된다. 이 측정은 기본적으로 0.1 이라는 최소값을 갖고 영상 i 및 j 에서의 각 블록의 합집합의 영역이 전체 겹치는 영역에 비해서 얼마의 비율을 가지고 있는지에 대한 값으로 결정된다. 최대값은 0.9 의 값을 갖지만 실제로 각 블록의 합집합의 영역이 절반 이상은 될 것이기 때문에 0.5 의 최대값을 가질 수 있다. 유사성을 측정해야 하는 영상 셋이나 실험 환경에 따라서 파라미터들은 변경될 수 있다.

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

$$\beta = 0.1 + 0.4 \times \left(1 - \frac{\sum_{k=1}^N \text{Region}_i(k)}{\text{Overlap_Region}_i}\right) + 0.4 \times \left(1 - \frac{\sum_{k=1}^N \text{Region}_j(k)}{\text{Overlap_Region}_j}\right) \quad (\text{식 } 6)$$

수식 6 에서 Overlap_Region_i 은 영상 i 에서의 겹치는 영역을 의미하고 $\text{Region}_i(k)$ 는 영상 i 에서 k 번째 블록의 영역을 의미한다.

3. 실험 결과

2 장에서의 컬러 유사성 측정은 한 채널일 경우에 대해서만 설명하고 있다. 영상이 그레이 스케일(grey-scale)일 경우에는 2 장에서의 측정 방법을 그대로 적용하면 된다. 영상이 RGB 컬러를 가지는 경우에는 식 3 에서의 S_{block} 을 RGB 각각에 대해서 구한 후에 모두 더한 값이 최종적인 블록 간의 비교를 측정하는 값이 되며 S_{image} 의 경우도 마찬가지이다. 그림 2(a)에 있는 두 영상 간의 컬러 차이가 어느 정도 존재하는 입력 영상에 대한 유사성 측정 결과는 표 1 과 같다.

표 1 컬러가 다른 두 영상의 유사성 측정

	R	G	B	RGB
S_{block}	2.152	2.393	2.456	7.002
S_{image}	0.330	0.405	0.106	0.842

표 1 에서 확인할 수 있듯이 두 영상을 전체적으로 비교하면 두 영상이 비슷하다고 할 수 있지만 대응되는 블록 간의 비교를 하면 그 차이가 더 커지는 것을 확인할 수 있다. 즉, 히스토그램을 구하거나 평균치를 구할 때 이용되는 샘플의 수가 적을수록 통계치는 더 의미를 가질 수 있으며 그 차이가 더 의미가 있어진다. 식 6 에 의해서 구해진 $\beta = 0.42$ 가 구해졌으며 이는 블록들이 포함하는 영상의 영역이 40% 정도라는 것을 확인할 수 있다. 최종적으로 구해진 $S_{\text{total}} = 7.355$ 이다.

컬러가 다른 두 영상을 컬러 보정을 한 후 영상의 유사성을 측정해 보았다. 두 영상의 컬러 보정은 두 영상의 컬러 변환 행렬을 구하는 방식[1]을 사용하였는데 컬러 패턴을 놓고 찍은 영상이 아닌 일반적인 장면을 찍은 영상을 사용하기 위해서 두 영상 간의 대응점을 검색하고 컬러 변환 행렬을 계산하였다. 컬러 보정 결과는 그림 6 과 같다.



그림 6 컬러 보정한 결과

표 2

	R	G	B	RGB
S_{block}	1.610	1.871	1.760	5.241
S_{image}	0.204	0.308	0.195	0.706

그림 6 에서 볼 수 있듯이 두 영상 사이의 컬러 차이는 줄어들었으며 표 2 에서와 같이 블록 간의 차이 및 영상 간의 차이에 대한 측정값 모두 표 1 에 비해서 줄어드는 것을 확인할 수 있다. 이 때 $\beta = 0.514$ 였으며, $S_{\text{total}} = 5.604$ 였다.

4. 결론

본 논문에서는 두 영상 간의 컬러 유사성 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 블록 기반의 특징량을 모두 사용하였다. 컬러 유사성 비교에서는 각 매칭점을 중심으로 한 블록 기반의 통계치 비교와 전체 영상에서의 통계치 비교를 하였으며 두 계산값 사이의 가중치는 블록 들이 전체 영상 영역 중에 얼마만큼 차지하는 지에 대한 것으로 결정하였다. 통계치로는 정규화된 평균의 차이와 히스토그램 차이를 사용하였다. 제안한 유사성 측정 방법은 다양한 컬러 유사성 측정에 사용될 수 있으며 특히 컬러 보정 전후의 컬러 유사성 비교에 적용가능함을 확인하였다.

참고문헌

- [1] Ilie, A.; Welch, G.; "Ensuring color consistency across multiple cameras," Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on , vol.2, no., pp.1268-1275 Vol. 2, 17-21 Oct. 2005
- [2] Yamamoto, K.; Kitahara, M.; Kimata, H.; Yendo, T.; Fujii, T.; Tanimoto, M.; Shimizu, S.; Kamikura, K.; Yashima, Y.; , "Multiview Video Coding Using View Interpolation and Color Correction," Circuits and Systems for Video Technology, IEEE Transactions on , 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 (1988). "A combined corner and edge detector". Proceedings of the 4th Alvey Vision Conference. pp. 147-151
- [5] Lowe, D. G., "Distinctive Image Features from Scale-Invariant Keypoints", International Journal of Computer Vision, 60, 2, pp. 91-110, 2004.
- [6] Bay, H., Tuytelaars, T., Gool, L.V., "SURF: Speeded Up Robust Features", Proceedings of the ninth European Conference on Computer Vision, May 2006
- [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, pages 384-396, 2002