

처리 속도 개선을 위한 공간 중복성 제거 기반 스테레오 매칭 알고리즘

*최민서 *최재영 *노용만

*한국과학기술원 영상비디오시스템 연구실

e-mail: choims@kaist.ac.kr

Time Efficient Stereo Matching Algorithm Using Spatial Redundancy Removal

*Choi, Min-Sheo *Choi, Jae Young *Ro, Yong Man

*Image and Video System Lab., Korea Advanced Institute of Science and Technology (KAIST)

요약

스테레오 매칭은 Robot Vision 분야에서 가장 활발하게 연구가 진행되어온 주제이다. 과거 많은 연구들은 스테레오 매칭 알고리즘의 정확도를 높이거나 처리속도를 개선하는 것 중 어느 한 쪽에만 집중하고 있었으며, 두 가지를 동시에 고려한 연구는 상대적으로 연구가 미흡한 실정이다. 본 논문에서는 영상의 공간적 중복 성질을 이용하여 스테레오 매칭 알고리즘의 처리 속도를 획기적으로 개선하는 방법을 제안하고, 처리 속도 감소에 따라 증가되는 disparity map 추정 오차를 보상하는 기술을 제안한다. 구체적으로 Dynamic Programming (DP) 기반 스테레오 매칭 알고리즘에 제안한 속도 개선 방법을 적용하고, 개선된 Adaptive Support Weight Filter를 사용하여 보다 효과적인 오차 보상을 실현 하였다. 체계적인 실험을 통해, 본 논문에서 제안 하는 스테레오 매칭 알고리즘은 처리속도와 정확도를 동시에 개선할 수 있음을 검증하였다.

1. 서론

스테레오 매칭(Stereo Matching 또는 Stereo Correspondence)은 Robot Vision 분야에서 가장 연구가 활발한 영역의 하나로서, 한 쌍의 2차원 스테레오 영상(즉, 좌영상과 우영상)으로부터 3차원 공간을 복원하거나, 또는 새로운 시점(View Point)의 영상을 생성하는 데 있어 가장 우선적으로 해결이 되어야 하는 문제 중 하나이다.[1]

Dynamic Programming (DP)은 실시간 스테레오 매칭을 위해 널리 사용되고 있는 글로벌 최적화 알고리즘의 하나로서 계산 자원이 제한된 환경에도 적용이 가능하다는 장점을 갖는다. 이러한 DP의 장점은 과거에 비해 강력해진 계산 자원을 갖춘 스마트 폰에서 증강현실 (Augmented Reality)이나 시점합성(New View Synthesis) 등의 응용 구현하기 위한 플랫폼으로 적용하기에 적합한 특성을 제공할 수 있을 것으로 기대되고 있다.

과거 DP를 활용한 스테레오 매칭을 방법들은 정확도를 개선하거나, 혹은 처리속도를 개선하기 위한 목적으로 수행되었다. 구체적으로 설명하면, 정확도는 알고리즘 수준에서 새로운 접근 방법에 의해 개선되어 왔으며 [2], [7], [8], [10], 속도 개선은 거의 대부분 하드웨어의 구조 변경이나 최적화 문제로 분리되어 연구되어 왔다 [4]. 이런 이유로 스테레오 매칭 알고리즘의 정확도와 처리속도를 동시에 개선하기 위한 연구는 매우 미흡하며 [4], [14], 더욱이 처리속도 개선을 알고리즘 수준에서 해결하려는 시도는 거의 찾아보기 어렵다. 또한, 이상과 같은 기존 방법들 한계는 알고리즘의 정확도를 개선시키는 경우에는 반드시 알고리즘 수준에서 처리속도의 증가를 수반한다는 것이다. 이는 실시간 스테레오 매칭 응용에서 많은 제약을 제공한다.

본 논문에서는 스테레오 매칭에서 정확도와 처리속도를 동시에 개선할 수 있는 효율적인 방안을 제안한다. 우선, 처리속도를 향상시키는 방법을 제시하고, 이를 DP 기반의 알고리즘에 적용하여 그 성능을 검증한다. 또한, 처리 속도에 감소로 야기되는 disparity map 추정에

관한 오차를 보상하기 위한 방법으로 효율적인 기존 필터링 방법을 이용하는 것은 물론 새로운 필터링 기법을 제안하여, 최종적으로 정확도와 처리속도를 동시에 모두 개선할 수 있는 방법은 제안한다.

본 논문의 구성은 다음과 같다. 제2장에서는 실시간 스테레오 매칭 알고리즘을 위한 속도 개선 방안에 대해 상술하고, 제3장에서는 기존의 Adaptive Support Weight Filter에 대한 개선 방안을 제시한다. 제4장에서는 DP 기반 알고리즘에 본 논문에서 제안한 방법을 적용한 몇 가지 구현예에 관한 실험결과를 보여준다. 마지막으로, 제5장에서 본 연구에 관한 결론을 도출한다.

2. 스테레오 매칭 알고리즘의 속도 개선 방안

가. 개요

그림 1 은 제안한 속도 개선방안에 따른 스테레오 매칭 알고리즘의 흐름도이다. 전형적인 스테레오 매칭 알고리즘에 비해 “Remove Spatial Redundancy”(공간적 중복성 제거) 과정이 추가된 점이 가장 큰 특징이며, 이 부분은 이하에서 상술한다.

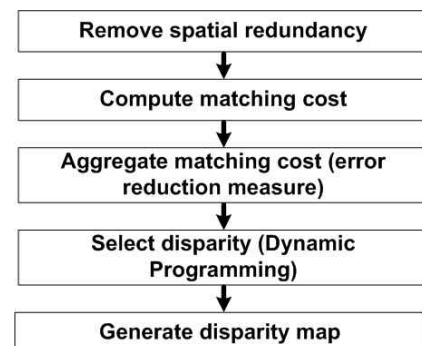


그림 1. 제안하는 스테레오 알고리즘 흐름도

나. 공간적 중복성 제거

그림 2에서 보이는 것 같이, 스테레오 매칭의 결과로 생성되는 깊이 영상(Depth Map 또는 Disparity Map)은 일반적으로 원 영상에 비해서 훨씬 더 단순하며 이에 따라 영상 내에 존재하는 공간적 중복성도 매우 크다. 따라서 스테레오 매칭을 위한 계산 과정에서 이러한 공간적 중복성을 효율적으로 제거한다면 상당한 계산 양과 프로세싱 시간 감소를 기대할 수 있다.



Left Image Right Image Disparity Map

그림 2. 스테레오 영상과 대응 깊이를 보여주는 예제 그림들

간단한 예비실험 결과를 통해 공간적 중복성 제거시 증가하는 에러율의 정도를 표 1에 나타내었다. 공간적 중복성 제거는 Ground Truth 영상에 다운샘플링을 적용한 후 다시 인터폴레이션(interpolation)을 통해 원영상의 크기로 복원하는 방법을 취하였다. 각각의 모드에서 수직방향으로 1/2 및 1/3 샘플링을 취했고, 마지막으로 수평 및 수직 방향에서 각각 1/2 씩 전체적으로 1/4의 비율로 다운샘플링을 실시하였다.

모드	Venus	Teddy	Cones
1/2	0.07%	1.14%	0.60%
1/3	0.26%	1.57%	0.92%
1/4 (1/2 x 1/2)	0.11%	1.15%	1.35%

표 1. 공간적 중복성 제거에 따른 에러율 증가를 나타내는 실험값

표 1에 나타낸 실험 결과들은 매우 단순한 방법으로 공간적 중복성을 제거하더라도 증가하는 에러율이 매우 낮은 수준임을 보여준다. 그러나 이러한 공간적 중복성 제거에 따라 감소되는 계산량은 상대적으로 매우 크다. 예컨대, 그림 3과 같이 수직방향으로 1/2 다운샘플링을 적용할 경우, 기존 DP 기반의 화소단위 스테레오 매칭 알고리즘에서 계산해야 할 매칭 코스트 어레이($C(x,y,d)$)는 다음의 식(1)과 같이 절반으로 줄어들게 되며, 이를 통해 적어도 이론상으로는 처리시간이 절반으로 단축될 수 있게 된다.

$$Size(C(x,y,d)) = [Width, \frac{1}{2}Height, DisparityRange] \quad (1)$$



그림 3. 수직방향의 1/2 다운샘플링의 예시 그림

한편, 이러한 다운 샘플링은 수직방향으로만 실시하는 것이 바람직한데, 이유는 수평방향의 다운샘플링은 깊이 해상도(Disparity Resolution)에 영향을 주기 때문이다. 예컨대, 1/N의 샘플링을 수평방

향으로 적용하는 경우, 깊이 해상도도 1/N으로 줄어들게 되어 스테레오 매칭의 결과로 생성되는 깊이 영상의 심각한 열화를 초래할 수 있기 때문이다. 따라서 수평방향의 다운샘플링의 사용여부는 적용분야의 특성을 고려하여 결정되어야 하며, 일반적으로 적용하기에는 무리가 있다.

다. 에러 보상 방법

앞서 살펴본 바와 같이 제안한 공간적 중복성 제거시에 발생하고 증가된 오차에 대처하기 위한 방안을 살펴보고자 한다. DP의 경우라면 다음의 두 가지 방안 정도를 고려할 수 있을 것이다. 첫째, 스캔라인 단위에서 독립적으로 이루어지는 DP 자체의 성능을 높이는 방법으로 예컨대 Salmen 등[8]이 제안한 다중 경로 DP를 이용할 수 있다. 둘째, 그림 1의 세 번째 단계(즉, Cost Aggregation)에서 수직방향으로 필터링을 하는 방법을 고려할 수 있다.

본 논문에서는 두 번째 방법에 기반 한 방법을 제안한다. 여기서 사용 가능한 필터는 계산량 증가가 미미한 간단한 가우시안 필터가 사용될 수 있으며, 비록 계산량이 크기는 하지만 Yoon 등[3]이 처음 제안하고 이후 Wang 등[4]이 DP에 처음 적용했던 Adaptive Support Weight Filter(ASWF)가 활용 될 수도 있다. 본 논문에서는 기존의 필터를 이용하는 대신에 효율적으로 오차를 보상하기 위해 다음 장에서 설명할 개선된 ASWF를 사용하는 방법을 제안한다.

3. Adaptive Support Weight Filter 활용 방법

그림 4는 제안한 ASWF의 개선방안을 설명하기 위한 것으로, A, B, C, D, E로 표시된 부분은 모두 수직방향 필터의 윈도우 범위를 나타내고 있다. 여기서 A, B, D, E는 상호간에 색상의 유사성이 높은 영역들이고, 이와 달리 영역 C는 다른 모든 영역과 비교하여 유사성을 가지고 있지 않다. 기존의 ASWF에서는 윈도우의 중심인 화소 A를 중심으로 Cost Aggregation을 행할 때, 거리와 색상 유사도에 기초한 가중치만을 고려하게 된다. 그러나 그림에서 영역 D와 E는 영역 C라는 불연속 구간을 거친 후에 존재하는 영역이므로 A와 동일 오브젝트에 속할 것이라는 가정의 Cost Aggregation은 수행하지 않는 것이 더 정확한 결과를 나타낼 수 있을 것이다. 요컨대, 제안하고자 하는 개선된 ASWF에서는 이러한 불연속 구간의 존재를 직접 판단하여 보다 정확한 Cost Aggregation이 이루어질 수 있도록 한다.

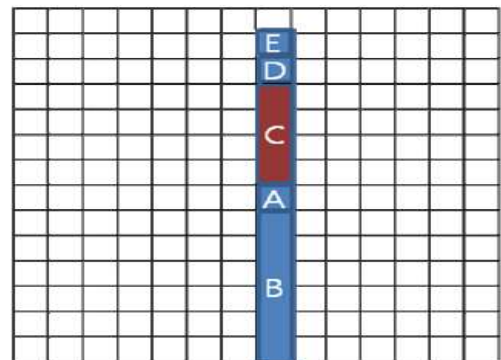


그림 4. 개선된 ASWF의 동작 원리

다음의 수학적 식 (2)는 제안한 필터에 따라 윈도우 내에 위치한 각각의 화소에 주어지는 가중치를 나타낸다.

$$w(p,l) = \begin{cases} \exp(-(\frac{\Delta C_{pl}}{r_c} + \frac{\Delta g_{pl}}{r_g})) & , \text{연속성 유지 구간} \\ 0 & , \text{그 외 구간} \end{cases} \quad (2)$$

여기서 r_c 와 r_g 는 각각 실험적으로 결정되는 가중치 상수이고, p 는 윈도우 내 중심 화소이고, l 은 윈도우 내에 위치한 중심 화소 이외의 모든 화소를 나타내며, ΔC_{pl} 과 Δg_{pl} 은 각각 색상의 차이와 유클리디안 거리를 나타낸다.

제한한 ASWF 필터에서 식 (2)에 따라 주어진 조정된 화소별 가중치는 마스크 매트릭스를 이용하여 미미한 계산량 증가만으로 구현이 가능하다.

4. 실험

제한한 방법의 유효성을 검증하기 위해 베이스라인(baseline) 모델로서 [13]에서 제안하는 방법을 참조하여 가장 간단한 형태의 DP를 구현하였다. 비교 방법 구현에서는 모두 수직방향의 1/2 다운샘플링을 통해 입력영상을 처리하되, Cost Aggregation 단계에서 서로 다른 필터를 사용하였다. 실험영상은 Middlebury 웹사이트 [15]에서 제공하는 가장 대표적인 4가지 영상인 Tsukuba, Venus, Teddy, Cones 등을 사용하였다.

본 실험에서는 총 5가지 구현 방법들을 구성하고 실행하여 비교 실험을 수행 하였다. 각각의 실험 방법에 구체적인 사항은 다음과 같다. “구현 방법 1”에서는 DP에 적용되는 파라미터를 더욱 세분화해서 사용하였고, “구현 방법 2”에서는 여기에 3×1 가우시안 필터링을 더 적용하였으며, “구현 방법 3”에서는 6×1 크기의 ASWF를 “구현 방법 4”에서는 12×1 크기의 ASWF를 각각 사용하였고, 마지막으로 “구현 방법 5”에서는 본 논문에서 제안하는 개선된 ASWF를 12×1 크기로 적용하였다.

표 2는 위에서 언급한 각각의 구현 방법에 따른 처리시간의 단축 정도를 나타낸 것으로, 처리시간은 베이스라인 모델에서 소요되는 시간을 100%로 하고, 각각 구현 방법에서 소요되는 시간을 상대적인 백분율로 표시하였다. 예상했던 바와 같이 구현 방법 1~5에서 상당한 처리시간 감소가 나타나고 있으며, ASWF와 같은 복잡한 필터를 사용한 경우에도 여전히 가장 간단한 DP보다 짧은 처리시간을 나타내고 있는 점은 특히 주목할 만하다.

구현 방법	처리시간
베이스라인	100%
구현 방법 1	48.3%
구현 방법 2	60.0%
구현 방법 3	60.7%
구현 방법 4	75.0%
구현 방법 5	80.3%

표 2. 각각의 구현 방법에 따른 처리시간 비교

구현 방법	Tsukuba	Venus	Teddy	Cones
베이스라인	14.7%	9.46%	15.3%	12.8%
구현 방법 1	11.3%	9.36%	16.9%	12.1%
구현 방법 2	9.93%	8.11%	17.6%	10.5%
구현 방법 3	5.21%	3.61%	11.2%	10.1%
구현 방법 4	5.75%	3.70%	11.9%	10.4%
구현 방법 5	3.97%	2.18%	10.3%	8.55%

표 3. 각각의 구현 방법에 따른 정확도(에러율) 비교

표 3은 각각의 구현예에 따른 정확도를 나타내고 있다. 간단한 가우시안 필터만 사용한 경우에도 처리시간을 40% 가량 줄이고도 정확도는 레퍼런스 모델을 능가하는 결과를 나타내었다. 또한, 여러 가지 필터링 방법 중 본 논문에서 제안하고 있는 개선된 ASWF가 가장 높은 정확도를 나타내었다. 표 3에 결과를 통해 제안하는 방법은 베이스라인 방법과 비교하여 disparity map추정을 위한 처리 시간을 상당히 줄이는 동시에 낮은 에러를 확보할 수 있는 것을 검증하였다.

5. 결론

본 논문에서는 실시간 스테레오 매칭 알고리즘에 적용 가능한 처리속도 개선 방법을 제안하였고, 실시간 알고리즘으로 가장 널리 사용되는 DP를 기반으로 제안한 방법의 유효성을 검증하였다. 또한 DP 기반의 알고리즘에서 수직방향 필터링 적용하여 정확도를 더욱 향상시킬 수 있도록 기존의 ASWF를 개선하기 위한 방안을 함께 제시하고 그 성능을 검증하였다. 본 논문에서 제안하는 방법은 실시간 스테레오 매칭을 요구하는 응용들에서 효과적으로 적용되는 것을 기대 할 수 있다.

참고문헌

- [1] D. Scharstein and R. Szeliski, A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms, IJCV 2002.
- [2] A. Criminisi, A. Blake, and C. Rother, Efficient Dense Stereo with Occlusions for New View-Synthesis by Four-State Dynamic Programming, International Journal of Computer Vision 71(10), 89-110, 2007.
- [3] K.-J. Yoon, and I.-S. Kweon, Adaptive Support-Weight Approach for Correspondence Search, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 28, No. 4, 650-656, 2006.
- [4] L. Wang, M. Liao, M. Gong, R. Yang, and D. Nistér, High-quality real-time stereo using adaptive cost aggregation and dynamic programming, 3DPVT 2006.
- [5] D. Scharstein, View Synthesis Using Stereo Vision, LNCS 1583, Springer, 1999.
- [6] S. Rogmans, J. Lu, P. Bekaert, and G. Lafuit, Real-Time Stereo-Based Synthesis Algorithms: A Unified Framework and evaluation on commodity GPUs, Signal Processing: *Image Communication* 24 (2009) 49-64.
- [7] J. C. Kim, K. M. Lee, B. T. Choi, and S. U. Lee, A Dense Stereo Matching Using Two-Pass Dynamic Programming with Generalized Ground Control Points, CVPR 2005 (2), 1075-1082
- [8] J. Salmen, M. Schlipfing, J. Edelbrunner, S. Hegemann, and S. Lueke, Real-Time Stereo Vision: Making more out of Dynamic Programming, CAIP 2009.
- [9] C. Lei, J. Selzer, and Y. Yang. Region-Tree Based

- Stereo Using Dynamic Programming Optimization, CVPR 2006.
- [10] O. Veksler, Stereo Correspondence by Dynamic Programming on a Tree, CVPR 2005.
 - [11] Y. Deng and X. Lin, A Fast Line Segment Based Dense Stereo Algorithm Using Tree Dynamic Programming, ECCV 2006.
 - [12] A. Klaus, M. Sormann and K. Karner, Segment-Based Stereo Matching Using Belief Propagation and a Self-Adapting Dissimilarity Measure, ICPR 2006.
 - [13] A. F. Bobick and S. S. Intille, Large Occlusion Stereo, IJCV 1999.
 - [14] S. Forstmann, Y. Kanou, J. Ohya, S. Thuring and A. Schmitt, Real-time stereo by using dynamic programming, CVPR 2004.
 - [15] <http://vision.middlebury.edu/stereo>