

Parallax와 움직이는 물체가 있는 비디오의 실시간 모자이크

정상일^o, 홍기상
 포항공과대학교 전자전기공학과
 {sijung, hongks}@postech.ac.kr

요약

본 논문에서는 수평이동 카메라로 획득한 비디오를 정합해 모자이크 영상을 생성하는 방법을 제안한다. 이 비디오 안에는 카메라가 장면을 보는 방향이 다름으로 해서 생기는 시차(parallax)와 걷는 사람과 같은 움직이는 물체가 존재한다. 제안된 방법은 이런 요소들로 인한 부정합이 없이 자연스러운 모자이크 영상을 생성한다. 기존의 방법들과 달리 순차적으로 정합하는 방식을 택함으로써 필요로 하는 메모리 영역을 줄였으며, 빠른 속도로 정합하여 실시간성을 확보하였다.

1. 서론

카메라가 가지는 시야는 한정되어 있다. 이런 한정된 시야를 넓히는 방법 중 가장 널리 알려진 모자이크(Mosaic) 방법은 좁은 시야를 가지는 여러 장의 영상들을 정합해 하나의 넓은 시야를 가지는 영상을 생성하는 방법이다.

기존의 일반적인 모자이크 방법들은 시차(parallax)가 없는 경우 모자이크 영상을 생성하는 방법들이다. 시차는 서로 다른 방향에서 평면이 아닌 장면(non-planar scene)을 바라보았을 때 생긴다. 시차가 없는 경우는 장면이 평면이거나 카메라의 움직임(motion)이 고정된 회전(rotation) 움직임일 경우에 해당된다. 또한, 장면(scene)이 카메라로부터 멀리 떨어져 있어서 있는 경우에도 카메라의 어느 정도 이동으로는 시차가 발생하지 않는다고 볼 수 있다. 두 영상을 정합한다고 가정했을 때 이런 경우는 두 영상 사이의 움직임(motion)이 하나의 호모그래피(homography)로 표현이 가능하다.

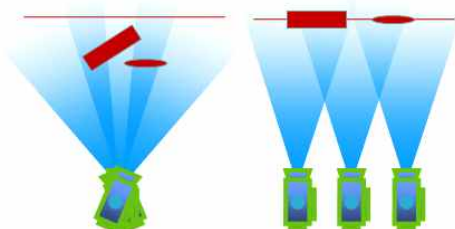


그림 1. 시차(parallax)가 없는 경우

최근에는 시차가 있는 입력 영상들로도 부정합이 없는 모자이크 영상을 생성하는 연구들이 진행되고 있다[1][2][3][4]. 시차가 존재한다는 것은 두 영상을 정합한다고 했을 때, 두 영상의 관계를 하나의 호모그래피(homography)로 나타낼 수 없다는 것을 뜻한다. 시차가 있는 영상들을 한 장의 영상으로 정합하는 것은 기하학적으로 문제가 발생한다. 이것을 해결하기 위해 지역적으로나마 실제 장면처럼 보이게 한다는 생각을 가지고 접근을 하고 있다. 시차뿐만 아니라 입력 영상들내의 움직이는 물체 또한 부정합의 원인이 된다. 시차와 마찬가지로 움직이는 물체 부분은 전체 영상의 움직임(global motion)과 차이가 있다. 즉, 하나의 호모그래피(homography)로 두 영상의 관계를 설명할 수가 없다.

<그림 2>에서 본 연구의 동기(motivation)을 도식적으로 표현하였다. 사용자가 핸드폰과 같은 저사양의 카메라로부터 동영상을 획득해 모자이크 영상을 생성한다고 했을 때, 시차와 움직이는 물체에 의한 부정합을 해결하는 것과 동시에 적은 메모리 공간과 빠른 처리 속도를 고려하여야 한다.

본 논문에서는 수평이동 카메라로 획득한 비디오로부터 모자이크 생성하는 방법을 제안한다. 비디오에는 시차와 움직이는 물체들이 존재하고, 이들로 인한 부정합이 없는 모자이크 영상을 생성한다. 기존의 방법과는 달리 적은 메모리 공간으로도 처리가 가능하도록 순차적(sequential)인 접근 방식을 채택하였고, 고속 처리로 실시간성을 확보하였다.

본 논문의 구성은 다음과 같다. 2장에서 제안된 순차적 비디오 모자이크 방법과 배경을 설명한다. 3장에서 실험 결과 및 기존 결과와의 비교를 통해 성능을 분석한다. 마지막으로 4장에서 본 논문이

제안하는 방법의 결론을 기술한다.

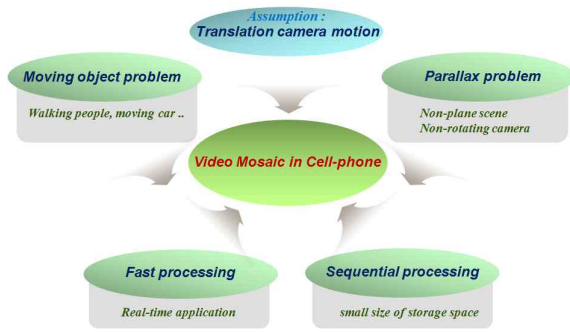


그림 2. 동기(motivation)

2. 제안된 순차적 비디오 모자이크 방법

2.1 개요

기존의 방법들이 시차와 움직이는 물체에 의한 부정합을 해결하기 위해 여러 정보를 이용하므로 처리 시간이 오래 걸린다. 또한, 일괄 처리 방식(batch processing)으로 인해 상당한 크기의 메모리 영역이 필요로 하게 된다. 핸드폰과 같은 휴대용 카메라로 비디오를 촬영해 모자이크 영상을 생성하고자 한다면 적은 메모리 공간에서 빠른 속도로 모자이크 영상을 생성하는 방식이 필요로 하는데 기존의 방식으로는 이것을 충족시킬 수 없다.

본 연구에서 제안하는 순차적 비디오 모자이크 방법의 전체적인 구성은 <그림 3>과 같이 구성되어 있다. 입력으로는 비디오 영상이 순차적으로 들어온다. 그것이 첫 번째 입력이면 그 자체가 모자이크 영상이며 가장 마지막으로 정합된 영상이다. 두 번째 영상이 들어오게 되면 첫 번째 영상과의 움직임 추정(motion estimation) 과정을 통해 두 영상간의 관계를 찾는다. 추정된 움직임(motion)을 바탕으로 들어온 입력 영상을 이전에 생성된 모자이크 영상에 일치(alignment)시킨다. 그 후 공통된 영역 내에서 두 영상을 가장 자연스럽게 정합시키는 경계(cut-line)를 찾아 두 영상을 정합시키면 두 번째 입력이 들어왔을 때까지의 모자이크 영상이 생성되게 된다. 이렇게 i 번째 입력 영상이 들어올 때마다 위와 같은 처리과정을 거쳐 모자이크 영상을 순차적으로 생성한다. 이런 방법은 기존 방법[2]과는 달리 전역적 일치(global alignment)를 통해 정합 경계(cut-line)를 찾는 영역(search range)를 줄여 처리 시간을 줄이는 효과를 가지며, 모든 들어온 영상들을 다 저장해 놓을 필요가 없이 마지막으로 정합된 영상과 모자이크 영상만 저장해 놓으면 되므로 필요한 메모리 영역이 크게 줄게 된다. 하지만, 시차와 움직이는 물

체가 존재하는 영상들이므로 두 영상의 전역 움직임(global motion)을 구하는 것이 하나의 문제가 된다. 2.2장에서는 두 영상의 전역 움직임 추정에 대해 기술하고, 2.3장에서는 전역 일치된 모자이크 영상과 붙여질 영상간의 정합 경계(cut-line)을 찾는 방법을 기술한다.

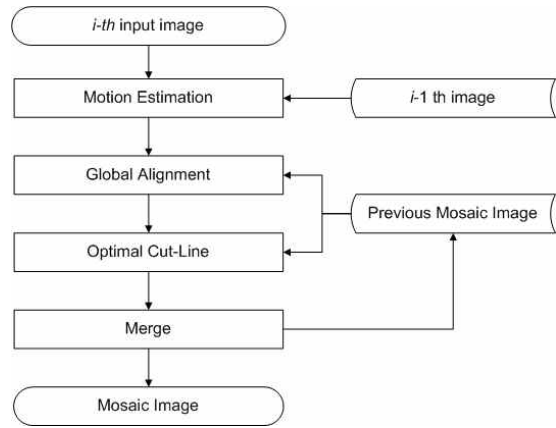


그림 3. 제안된 시스템의 전체적인 구성

2.2 움직임 추정(Motion Estimation)

서론에서 설명하였듯이 시차와 움직이는 물체가 존재하는 두 영상간의 관계는 하나의 호모그래피(homography)로 나타낼 수 없다. 본 연구에서는 전역적 일치(global alignment)를 배경(background) 움직임에 맞추었다. 배경을 일치 시켜 그 부분에서 정합 경계를 찾아 모자이크 이미지를 생성하면 시차가 발생하는 부분과 움직이는 물체가 있는 부분에서 정합 경계가 발생해 영상 왜곡이 발생하는 것을 차단할 수 있다. 모자이크 영상은 여러 장의 영상이 계속 합쳐져 변형되므로 붙여질 영상과의 움직임이 새로운 모자이크 영상을 생성하는데 문제로 작용할 수 있다. 따라서 확장되는 영역의 움직임을 맞추기 위해 이전 모자이크 영상이 아닌 마지막에 정합된 영상과 새로 들어온 영상의 움직임을 추정하여 전역 일치를 시킨다.

움직임 추정은 multi-resolution block matching algorithm(MR-BMA)[5]을 사용하였다. 이 알고리즘은 3단계의 해상도의 영상들을 생성하여 블록 매칭(block matching)을 coarse-to-fine 방식으로 수행하여 기존의 전영역의 조사(full search)하는 방식보다 블록 매칭의 속도가 매우 빠르다. 이렇게 대응된 점들(corresponding points)을 가지고 최소자승법(least square)을 통해 두 영상간의 움직임(motion)을 구한다. 이렇게 구한 움직임은 시차가 있는 부분이나 움직이는 물체가 있는 부분에서 구한 대응점들도 이용하였으므로 전체적으로 잘 일치가 되지 않는

다. 따라서 이 부분들에 해당하는 대응점들을 이상점(outlier)으로 고려하여 제거하고 배경에서 얻은 대응점들만으로 움직임을 추정해 최종 전역 움직임(global motion)을 얻게 된다. 최초로 얻은 움직임과 실제 대응점의 움직임과의 차이를 가지고 히스토그램(histogram)을 생성하여 작은 쪽 50%의 대응점만을 배경 부분에 해당하는 대응점으로 고려한다.

카메라가 수평이동을 하므로 움직임은 x,y방향의 움직임(2-parameters)만 고려하여 추정하였다. 장면이 실제 같은 형상을 갖기 위해서는 변형이 최대한 고정(rigid)되어야 한다[6][7]. 즉, x,y,θ의 움직임만을 추정하여 변형하는 것이 이상적이다. 본 논문에서는 회전 움직임을 고려하지 않았으나 추후 이 움직임도 고려하여 보다 일반적인 움직임에도 안정된 모자이크 영상을 생성하는 방법을 고안하고자 한다.

2.3 최적의 정합 경계(Optimal Cut-Line)

움직임 추정을 통해 이전 모자이크 영상과 붙여질 영상을 전역 일치시킨 후 두 영상을 이어 자연스럽게 새 모자이크 영상을 얻기 위해서 최적의 정합 경계를 찾는 것이 필요하다. <그림 4>와 같이 공통된 영역의 각 열을 하나의 정점(node)으로 간주하여 이전 모자이크 영상의 한 정점을 전역 일치된 붙여질 영상의 정점과 간선(edge)로 연결한다. 움직임 추정이 1 픽셀(pixel) 이내의 오차를 갖는다고 가정하여 전역 일치된 정점의 좌우 한 개씩의 정점을 더 연결하여 이전 모자이크 영상의 한 정점 당 총 3개의 간선을 생성한다. 각 간선의 비용(cost)은 각 정점 주변의 유사도 측정(similarity measure)를 통해 주어지고 유사도가 클수록 작은 간선 비용을 준다. 이 중 가장 작은 비용(minimum cost)을 가지는 간선에서 천이(transition)가 일어나게 한다면 두 영상을 가장 자연스럽게 붙일 수 있다.

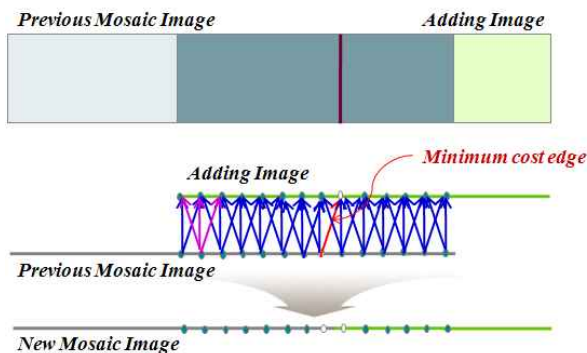


그림 4. 최적의 정합 경계

유사도 측정(similarity measure)은 <그림 5>와 같이 두 정점에 해당하는 열(column) 주변의 유사도를 계산한다[2]. 이전 모자이크 영상의 i 번째 정점에 해

당하는 열을 f_i 로 명명하고, 새로 붙여질 영상의 j 번째 정점에 해당하는 열을 g_j 로 명명한다. f_i 를 포함한 이전 k 개의 열을 하나의 줄영역(strip) ψ_f^i 라 정의하고 f_i 이후 k 개의 열을 줄영역 ψ_f^{i+1} 로 정의한다. 마찬가지로 g_j 주변의 줄영역도 이와 같이 정의하여 g_j 를 포함한 이후 k 개의 열을 줄영역 ψ_g^j 라 정의하고 g_j 이전 k 개의 열을 줄영역 ψ_g^{j-1} 로 정의한다. 두 정점 f_i 와 g_j 를 연결하는 간선의 비용을 다음과 같이 정의한다.

$$C(f_i, g_j) = \min \{ \| \psi_g^j - \psi_f^{i+1} \|, \| \psi_f^i - \psi_g^{j-1} \| \} \quad (1)$$

여기서 $\| \psi_g^j - \psi_f^{i+1} \|$ 와 $\| \psi_f^i - \psi_g^{j-1} \|$ 는 다음과 같이 정의한다.

$$\| \psi_g^j - \psi_f^{i+1} \| = \max \left\{ \begin{array}{c} \| \psi_g^j - \psi_f^{i+1} \|_{seg1} \\ \vdots \\ \| \psi_g^j - \psi_f^{i+1} \|_{segn} \end{array} \right\} \quad (2)$$

$$\| \psi_f^i - \psi_g^{j-1} \| = \max \left\{ \begin{array}{c} \| \psi_f^i - \psi_g^{j-1} \|_{seg1} \\ \vdots \\ \| \psi_f^i - \psi_g^{j-1} \|_{segn} \end{array} \right\} \quad (3)$$

즉, $\| \psi_g^j - \psi_f^{i+1} \|$ 는 두 줄영역 ψ_g^j 와 ψ_f^{i+1} 의 직접적인 SAD(sum of difference)가 아니라 두 줄영역을 n 개의 작은 영역으로 나누어 SAD를 구한 후 최대 SAD로 정의한다. 모자이크 영상에서 사람의 눈에 띄는 정합 경계는 선이 끊어지거나 부분적으로 자연스럽게 연결되지 않는 곳에서 일어난다. 따라서 위의 조건은 전체적인 차이가 아니라 지역적으로 얼마나 자연스러운가를 잘 나타내는 측정량(measure)이다. 일반적으로 SSD(sum of squared difference)를 사용하여 유사도 측정을 하여 작은 차이에도 큰 간선 비용을 부여한다. 하지만 제안된 방법은 최소 비용을 가지는 단 하나의 간선만 찾으면 되므로 SAD를 사용함으로써 모자이크 영상의 질

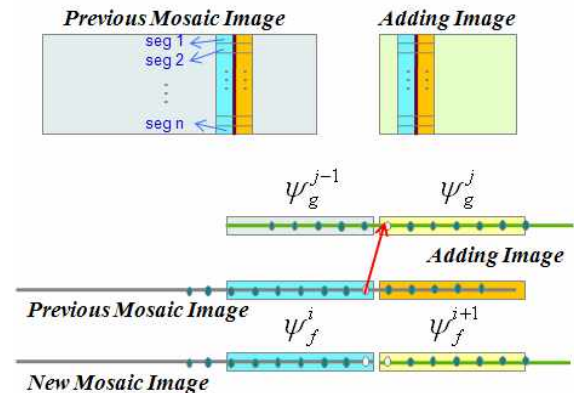


그림 5. 유사도 측정(similarity measure)

을 저하시키지 않으면서 처리 시간을 줄였다.

또한 불필요하게 중복된 계산을 줄여 시간복잡도를 낮추었다. <그림 6>은 어떤 부분에서 중복된 계산이 일어나는지 보여준다.

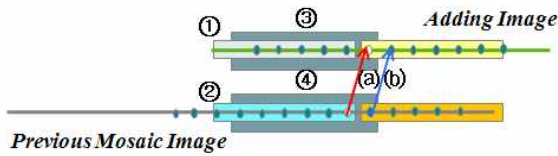


그림 6. 중복된 간선 비용 계산

간선 (a)의 비용을 계산하기 위해서는 줄영역 ①과 ②의 SAD를 계산해야 한다. 마찬가지로 간선 (b)의 비용을 계산하기 위해서는 줄영역 ③과 ④의 SAD를 계산해야 한다. 즉, 겹쳐지는 영역에 중복된 SAD 계산이 일어난다. 이를 해결하기 위해 적분 영상(integral image)[8]을 이용하였다. <그림 7>에 그 과정을 나타내었다. 공통된 영역에 대한 절대차(absolute difference) 영상의 적분 영상을 생성한다. A,B,C,D로 둘러싸인 영역의 SAD를 구하고자 한다면 그 안의 모든 픽셀들의 절대차를 다시 계산할 필요 없이 영역의 네 모퉁이 값만 적분 영상에서 가져와 계산하면 쉽게 구할 수 있다.

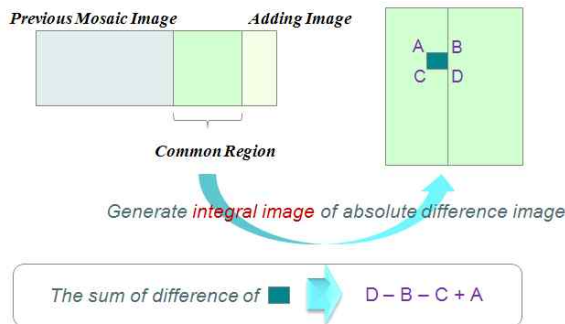


그림 7. 적분 영상을 이용한 중복된 계산 회피

3. 실험 결과 및 분석

실험 평가는 본 논문에서 제안한 방법에 의해 생성된 모자이크 영상 결과와 기존 방법[2]으로 생성한 결과를 비교 평가한다. 제안된 방법의 시스템은 CPU 2.66GHz, RAM 3.25GHz 하드웨어 환경에서 C++로 구현하였다.

3.1 실험 결과

<그림 8>은 시차가 크게 존재하는 비디오 영상의 모자이크 결과를 보여준다. 제안된 방법으로 생성

한 모자이크 결과가 기존의 방법으로 생성된 모자이크 결과와 거의 차이가 없음을 알 수 있다. 이 영상은 360x240 (pixel)의 크기를 가지는 147개의 프레임으로 구성된 영상으로 기존의 방법으로는 거의 1~2분 정도의 시간이 걸리나 제안된 방법으로는 약 3초 정도의 시간이 걸림으로써 약 50(frames/sec)의 속도 성능을 나타내었다. <표 1>에서 <그림 8>의 영상과 <그림 9>의 영상에 대한 영상 속도를 볼 수 있다. <그림 9>의 영상(영상크기:360x219)에 대한 속도도 45.4 (frames/sec)로 우수한 성능을 보인다.

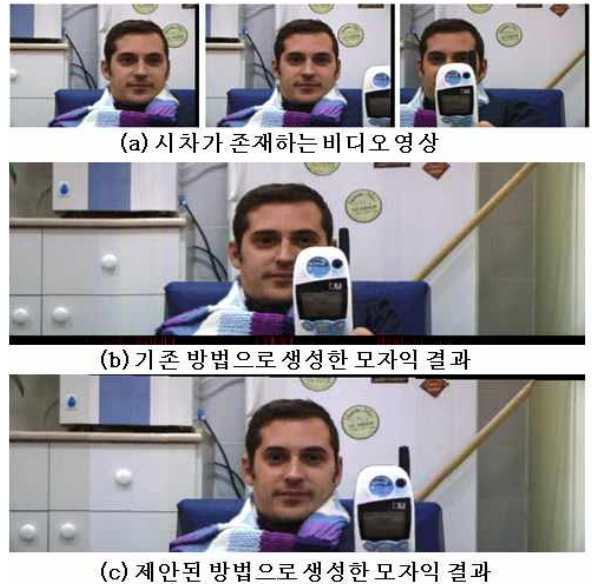


그림 8. 모자이크 결과 비교 ('cellphone')

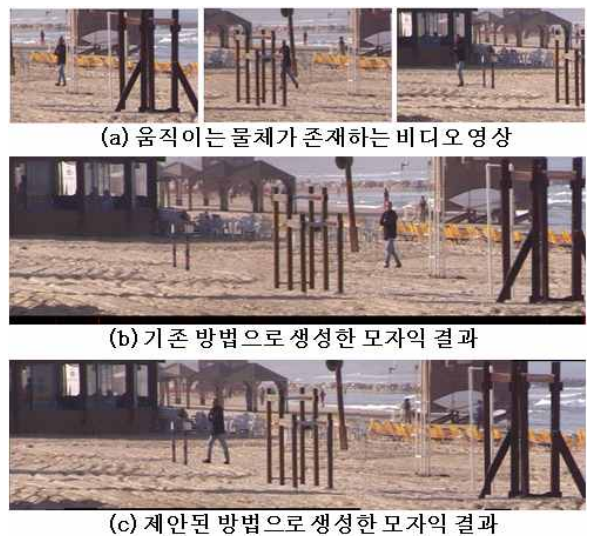


그림 9. 모자이크 결과 비교 ('beach')

<그림 9>는 움직이는 물체가 존재하는 영상에 대한 모자이크 영상 생성 결과이다. 부정합 부분이 없이 잘 정합되었음을 확인할 수 있다.

표 1. 모자이크 영상 생성 속도

Video	Frame rates (frames/sec)	
'cellphone'	Motion estimation	129.9
	Optimal cut-line	139.2
	Total	50.5
'beach'	Motion estimation	93.5
	Optimal cut-line	159.2
	Total	45.4

3.2 한계점

제안된 방법으로 모자이크 영상을 생성할 때 수평 이동 카메라로부터 촬영된 영상에 손떨림(jitter)이 있을 경우에도 잘 동작한다. 이것은 y방향의 움직임까지 고려하여 전역 일치(global alignment)를 시키기 때문이다. 하지만 손떨림이 있는 상태에서 수직 방향으로의 시차가 동시에 존재하거나 회전 움직임이 동시에 존재하는 경우는 정확히 작동하지 않는다. 또한 움직이는 물체가 커서 움직임 추정이 움직이는 물체에 맞춰지는 경우에도 부정합이 일어나는 경우가 생긴다. <그림 10>은 이런 경우의 모자이크 결과를 보여준다.



(a) 손떨림(jitter)와 수직방향의 시차가 동시에 존재하는 경우



(b) 움직이는 물체에 의해 움직임 추정이 정확하지 않을 경우

그림 10. 부정합이 있는 모자이크 영상

4. 결론

본 논문에서는 수평이동 카메라로부터 획득한 비디오 영상에 시차와 움직이는 물체가 존재할 때 부정합이 없는 모자이크 영상을 순차적으로 빠르게 생성하는 방법을 제안하였다. 이 방법은 기존의 방

법으로 생성한 모자이크 결과와 비슷한 결과를 나타내면서도 시간 복잡도(time complexity)를 줄여 360x240 정도의 크기를 가지는 영상에서 약 45~50 (frames/sec)의 처리 속도를 가짐으로써 실시간성을 확보하였다. 또한 순차적 처리 방식을 사용함으로써 필요한 메모리의 크기도 상당히 줄이는 효과를 나타내었다.

다만, 손떨림이 존재할 때 또는 회전 움직임이 존재할 때에 시차가 동시에 발생한다면 부정합이 발생하는 한계점을 보였다. 카메라의 움직임은 사람이 조정하므로 이런 움직임에도 좋은 모자이크 영상을 생성하는 연구가 진행되어야 한다.

참고문헌

- [1] J. Jia and C.K. Tang, "Eliminating Structure and Intensity Misalignment in Image Stitching," IEEE International Conference on Computer Vision, vol. 2, pp. 1651-1658, 2005.
- [2] Y. Wexler and D. Simakov, "Space-Time Scene Manifolds," IEEE International Conference on Computer Vision, vol. 1, pp. 858-863, 2005.
- [3] A. Rav-Acha, Y. Shor, and S. Peleg, "Mosaicing with Parallax using Time Warping," Proceedings of Conference on Computer Vision and Pattern Recognition, vol. 11, pp. 164, 2004.
- [4] A. Rav-Acha, G. Engel, and S. Peleg, "Minimal Aspect Distortion (MAD) Mosaicing of Long Scenes," International Journal of Computer Vision, vol. 78, no. 2-3, pp. 187-206, 2008.
- [5] J.H. Lee, K.W. Lim, B.C. Song, and J.B. Ra, "A Fast Multi-Resolution Block Matching Algorithm and its LSI Architecture for Low Bit-Rate Video Coding," IEEE Transactions on Circuits and Systems for Video Technology, vol. 11, no. 12, pp. 1289-1301, 2001.
- [6] M. Alexa, D. Cohen-Or, and D. Levin, "As-rigid-as-possible Shape Interpolation," International Conference on Computer Graphics and Interactive Techniques, pp. 157-164, 2000.
- [7] T. Igarashi, T. Moscovich, and J.F. Hughes, "As-rigid-as-possible shape manipulation," ACM Transactions on Graphics, vol. 24, no. 3, pp. 1134-1141, 2005.
- [8] P. Viola and M.J. Jones, "Robust Real-time Face Detection," International Journal of Computer Vision, vol. 52, no. 2, pp. 137-154, 2004.