**Super SloMo: High Quality Estimation of Multiple Intermediate Frames for Video Interpolation**

**1.** **Introduction**

기존의 비디오 interpolation은 주로 재귀적인 단일 프레임 비디오 interpolation에 초점을 맞추고 있어서 전의 프레임이 생성이 되어야 다음 프레임이 생성이 되기 때문에 속도가 느리고, 원하는 프레임수을 생성할 수 없었다(개의 프레임만 생성). 이 논문에서는 가변 길이 멀티 프레임 interpolation을 제시한다.

**2. Related Work**

**Video interpolation.**

Mahajan et al : 이미지 gradients를 지정된 시간 단계로 이동하고 Poisson 방정식을 해결하여 interpolation 프레임을 생성. but 멀티 중간 프레임을 생성할 수 있지만 복잡한 최적화 문제로 인해 계산량이 많음.

*Meyer et al* : 비디오 interpolation을 위해 지향적인 멀티스케일 피라미드 수준에 걸쳐 단계 정보를 전파할 것을 제안. but 움직임이 큰 high-frequency컨텐츠에는 여전히 실패하는 경향이 있음.

*Long et al* : optical flow에 대한 CNN모델을 배우기 위한 지도 신호로 프레임 interpolation을 사용. but 주요 목표는 optical flow이며 interpolation 프레임은 흐릿한 경향이 있음.

*Niklaus et al* : 프레임 interpolation을 두개의 입력 프레임에 대한 로컬 convolution으로 간주하고 각 픽셀에 대해 spatially-adaptive convolution 커널을 배우기 위해 CNN을 사용. but 모든 픽셀에 대한 커널을 예측하는 것은 계산이 많고 메모리도 많이 필요.

*Niklaus et al* : 분리 가능한 커널을 예측하여 효율을 개선. but 처리할 수 있는 모션은 커널 크기에 의해 제한됨(up to 51 pixels).

*Liu et al* : 모션 추정을 위한 명백한 서브 네트워크를 가진 프레임 interpolation을 위한 CNN모델을 개발. but 이러한 CNN기반 단일 프레임 interpolation은 다중 프레임 interpolation에는 적합하지 않다.

Wang et al: 다른 표준 카메라에서 가져온 비디오 프레임을 참고 자료로 사용하여 라이트 필드 비디오의 중간 프레임을 생성하기 위해 조사 but 우리의 방법은 평범한 비디오용 중간 프레임을 만드는 것을 목표로 하고 있고 참조 이미지가 필요하지 않음.

**Learning optical flow**

*Horn and Schunck* : 변형 접근법을 도입(<http://www.navisphere.net/2233/determining-optical-flows/> ). 피쳐 매칭은 종종 작고 빠르게 움직이는 물체를 다루기 위해 도입됨. but 복잡한 활성 함수의 최적화를 요구하고, 계산량이 많다.

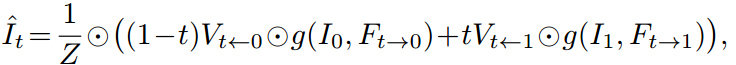
*Dosovitskiy et al* : FlowNetS와 FlowNetC라는 두개의 네트워크 아키텍쳐를 개발하고, CNN모델을 사용하여 두개의 입력 영상에서 optical flow로의 매핑을 학습하는 것의 가능성을 보여줌.

*Ilg et al* : 훨씬 더 나은 성능을 얻기 위해 더 큰 네트워크인 FlowNet2를 설계하기 위해 FlowNetS와 FlowNetC를 빌딩블록으로 사용. 최근의 두가지 방법 또한 네트워크 아키텍쳐의 optical flow의 기존 원리를 구축하고 비교 가능하거나 심지어 더 나은 결과를 달성하며 FlowNet2보다 더 적은 연산을 요구하도록 제안됨. 입력 이미지중 하나를 다른 이미지로 옮기기 위해 예측된 흐름을 사용. reconstruction error는 네트워크를 학습하는 지도 신호 역할을 한다. 단지 두개의 프레임만 고려하는 대신, 메모리 모듈은 비디오 시퀀스의 시간적 정보를 유지하도록 제안.

Liang et al: 비디오 프레임 추정을 통해 optical flow를 학습하지만, 그들의 학습은 추가적인 지도 신호로써 EpicFlow방법에 의해 추정된 흐름을 이용.

**3. Proposed Approach**

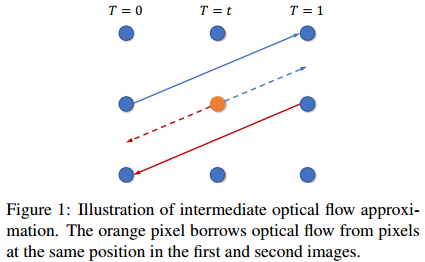
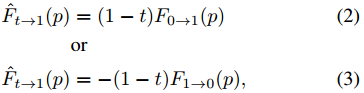
**3.1. Intermediate Frame Synthesis**, 은 각각 →, →로 가는 optical flow을 나타낸다. 이 두 흐름의 필드를 알게 되면 다음과 같이 중간 이미지 를 합성할 수 있다.

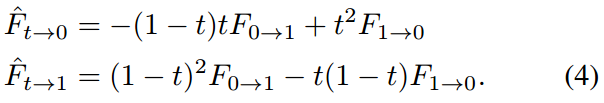
g(o,o)는 backward warping함수로서, bilinear interpolation를 사용해서 시행되고, 미분할수있다. 는 두 입력 영상의 영향를 제어하며 두가지 요인인 temporal consistency와 occlusion reasoning에 따라 달라진다. ⊙는 요소별 곱셈을 의미하며, 입력이미지의 내용인식 가중치를 의미한다. T=t가 T=0에 가까울수록 가 에 더 많은 기여를 하고 에 대해서도 유사한 속성이 적용된다. 비디오 프레임 interpolation 문제의 중요한 속성은 픽셀 p가 T=t에서 보이는 경우 적어도 입력 영상 중 하나에서 보일 가능성이 높으며 occlusion 문제를 해결할 수 있음을 의미한다. 따라서 우리는 visibility maps , 를 도입한다. (p)∈[0,1]은 T=0에서 T=t로 이동할 때 픽셀 p가 가시적인 상태로 유지되는지 여부를 나타낸다. temporal consistency와 occlusion reasoning을 결합한 식이

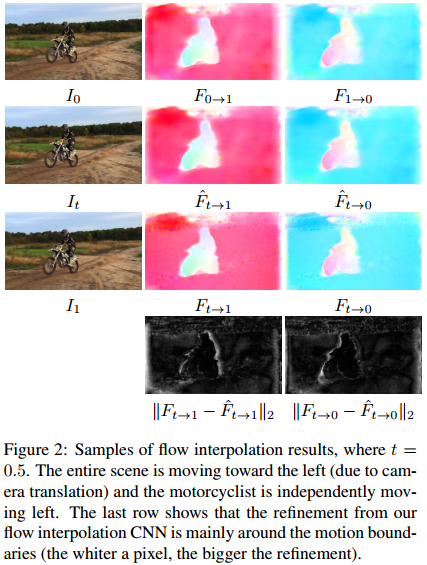
이고 여기서

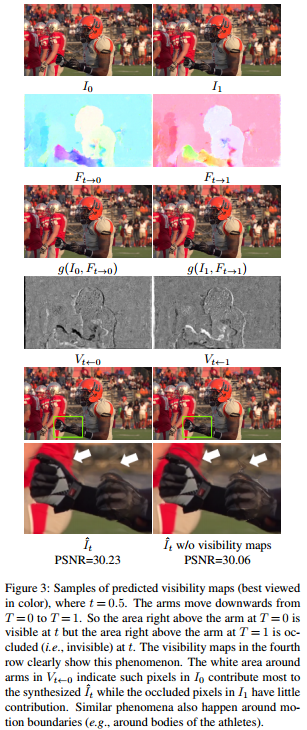
는 normalization factor이다.

**3.2. Arbitrary-time Flow Interpolation**

흐름 필드 , 를 계산하기 위해 우리는 두개의 입력 이미지 , 사이의 optical flow을 사용하여 중간 optical flow , 를 대략적으로 합성할 수 있다. T=t에서 주황색 점 p의 경우, 우리는 T=1에서 해당하는 픽셀에 optical flow을 합성하려 한다(파란 점선). 한가지 간단한 방법은 optical flow이 로컬적으로 스무스하다고 가정할 때 T=0, T=1에서 동일한 격자 위치에서 optical flow을 빌리는 것이다(파란색, 빨간색 실선). 특히 (p)는 수식(2)또는 수식(3)으로 근사치를 구할 수 있으며, 여기서 동일한 방향 또는 반대 방향으로 두 입력 이미지 사이의 optical flow 방향을 취하고 그에 따라 크기를 조정할 수 있다(수식(3)중의 (1-t)).

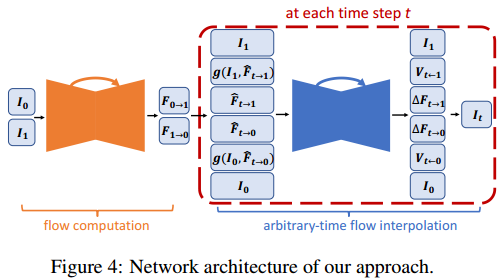
RGB이미지 합성의 시간적 일관성과 유사하게, 다음과 같이 양방향 입력 optical flow을 결합하여 중간 광학적 흐름을 근사하게 만들 수 있다(벡터형태). 

이 근사치 작업은 스무스한 지역에서 잘 작동하지만 모션 경계선 주변은 스무스 하지 않기 때문에 결과가 형변없다. 모션 합성 불량으로 이어질 수 있는 모션 경계 주변의 아티팩트를 줄이기 위해 초기 근사치를 세분화하는 방법을 학습할 것을 제안한다. 흐름 interpolation 서브 네트워크를 학습한다. 이 서브네트워크는 입력 이미지 , , 그리고 그것들 사이의 optical flow, , 흐름 근사치 , , 흐름근사치를 사용한 변형된 두 입력 이미지g(,), g(,)를 사용해서, 개선된 중간 optical flow 필드 , 을 출력한다.

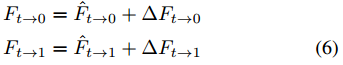
Visibility maps은 occlusions을 다루기 위해 필수적이다. 따라서, 우리는 흐름 interpolation CNN을 이용하여 두개의 visibility maps , 를 예측하고, (5)를 만족시키도록 한다.

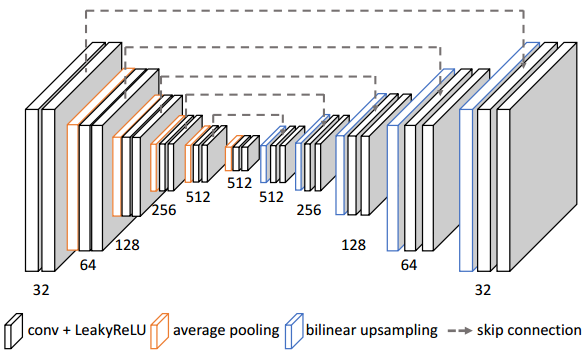
직관적으로 (p)=0은 (p)=1을 의미하고, 의 픽셀 p가 T=t에서 차단되므로 을 완전히 신뢰해야 하며 그 반대도 마찬가지이다. 0과 1 동시에 픽셀이 차단되는 경우는 거의 없다는 점에 유의한다. 픽셀 p가 , 에서 보두 보일 때 soft visibility maps을 사용하기 떄문에, 네트워크는 두 이미지에서 얻은 정보를 매팅 효과와 유사하게 적응적으로 결합하는 법을 배운다.

흐름 interpolation을 시행하려면 먼저 두 입력 이미지 사이의 양방향 optical flow을 계산해야 한다. 이 논문에서는 두개의 입력이미지 , 을 취하여 그 사이에 전방 optical flow 과, 후방 optical flow 을 같이 예측하도록 흐름 연산 CNN을 학습한다.



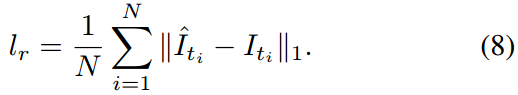
Flow computation 및 흐름 interpolation CNN을 위해 U-Net 아키텍쳐를 사용한다. U-Net은 fully convolutional neural network이고, 동일한 공간 해상도에서 인코더와 디코더 피쳐간의 skip connection을 사용하는 인코더와 디코더로 구성된다. 인코더에는 두개의 convolutional layer와 하나의 leaky ReLU (a = 0.1)인 layer로 구성되는 6개의 계층구조가 있다. 마지막 layer을 제외한 각 layer의 끝에서 stride가 2인 average pooling layer가 공간 치수를 감소시키기 위해 사용된다. 디코더 부분에는 5개의 layer가 있다. 각 layer의 시작부분에서 2배만큼 공간 치수를 증가시키기 위해 bilinear upsampling layer를 사용하고 그 다음에는 2개의 convolution과 Leaky ReLU layer이 사용된다. Flow computation CNN의 경우, 큰 움직임을 포착하기 위해 인코더의 처음 몇 개 layer에 큰 필터를 설치하는 것이 중요하다. 따라서 첫 두개의 convolutional layer에는 7x7 커널을 사용하고, 두번째 층에는 5x5를 사용한다. 나머지는 3x3 커널을 사용한다. 우리는 흐름 interpolation 네트워크의 디코더의 입력으로 2개의 네트워크 안의 인코더의 출력을 같이 결합하는 것이 조금 더 좋은 결과를 낸다는 것을 발견했다. 또한 중간 optical flow을 직접 예측하는 대신 중간 optical flow의 잔여(residual)을 예측하는 것이 약간 더 낫다는 것을 알게 되었다. 구체적으로 흐름 interpolation 네트워크는 와 을 예측한다.





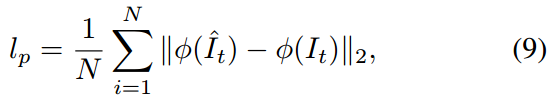
**3.3. Training**

주어진 입력 이미지 , 사이의 중간 프레임의 집합 (∈ (0, 1)), 및 중간 프레임의 예측을 고려할 때 loss function은 

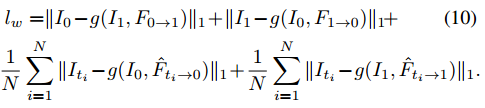
중간 프레임의 복원이 얼마나 좋은지 모델링하는Reconstruction loss 은

이러한 Reconstruction loss는 픽셀 값이 [0,255]안에 있는 RGB공간에 정의된다.

**Perceptual loss.** L1 loss를 중간 프레임의 복원 에러를 모델링하는 데 사용하지만, 예측이 여전히 흐릿한 결과를 초래할 수 있다. 따라서 예측의 세부사항을 보존하고 interpolation프레임을 더 선명하게 만들기 위해 perceptual loss를 사용한다.



여기서 ∮는 ImageNet으로 사전 학습된 VGG16모델의 conv4\_3 피쳐를 말한다.

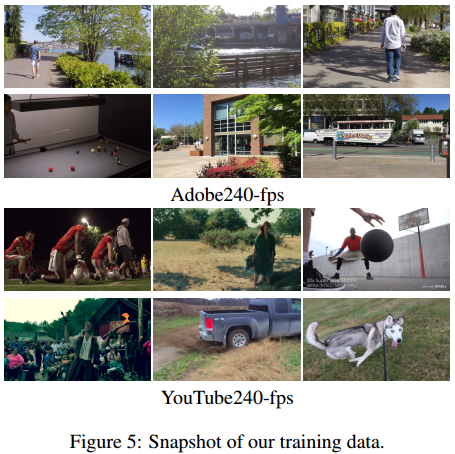
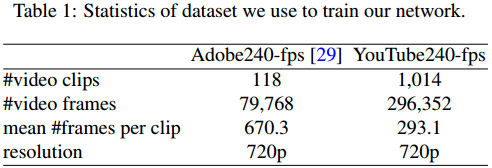
**Warping loss.** 중간 예측 외에도 수식(10)처럼 정의되는 계산된 optical flow의 품질을 모델화하기 위한 warping loss 를 도입한다.

**Smoothness loss.** 마지막으로 주변 픽셀에 유사한 흐름 값을 가지도록 smoothness term을 추가한다. 

가중치는 =0.8,=0.005, =0.4, =1를 사용하여 경험적으로 설정되었다. 뒤틀림과 흐름 근사치를 포함한 네트워크의 모든 요소들은 미분할 수 있다. 따라서 우리의 모델은 end-to-end 학습을 할 수 있다.

**4. Experiments4.1. Dataset**

우리의 네트워크를 학습하기 위해, 소형카메라로 찍은 [29]의 240fps 비디오를 사용한다. 또한 유튜브에서 240fps의 비디오 데이터셋을 수집한다.



총 1132개의 비디오 클립과 376k개의 개별 비디오 프레임이 있다. 우리가 가지고 있는 모든 데이터로 네트워크를 학습하고, Middlebury benchmark, UCF101, slowflow dataset, high-frame-rate Sintel sequences들을 포함한 여러 개의 독립적인 데이터셋으로 모델을 테스트한다.

Middlebury에서 우리는 8개 시퀀스의 단일 프레임 비디오 interpolation 결과를 평가 서버에 제출한다.

UCF101의 경우, 프레임의 3개마다 [15]에서 제공한 379 시퀀스를 사용해 두번째 프레임을 예측하기 위해, 첫 번째 프레임과 세 번째 프레임을 입력으로 사용한다.

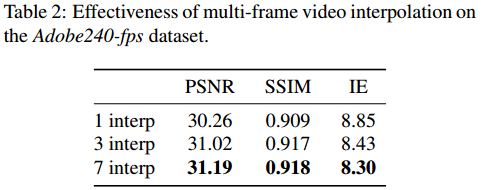
Slowflow dataset에는 전문 고속 카메라로 촬영한 46개의 비디오가 포함되어 있다. 우리는 첫번째와 여덟번째 비디오 프레임을 입력으로 사용하고, 30fps비디오를 240fps비디오로 변환하는것처럼 중간7프레임을 interpolation한다.

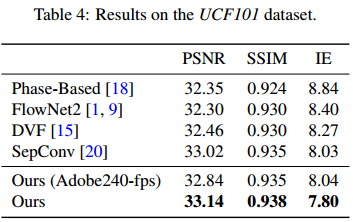
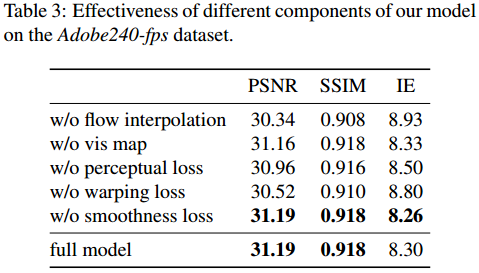
Original Sintel 시퀀스는 24fps로 랜더링되었다. 그 중 13개가 1008fps로 리랜더링 되었다. 우리는 이전의 방법들과 공정한 비교를 위해 중간 프레임 31개만 예측한다().

500 epoch동안 Adam optimizer를 사용. Learning rate는 0.0001로 초기화하고 200에폭마다 10배씩 감소. 학습 중에 모든 비디오 클립은 먼저 12프레임의 짧은 클립으로 나누어지며, 어느 것도 겹치지 않는다. 데이터 증가를 위해 전체 시퀀스의 방향을 랜덤하게 반전시키고 학습을 위해 9개의 연속된 프레임을 선택한다. 영상 레벨에선, 각 비디오 프레임은 360보다 짧은 공간치수를 가지도록 조정하고 352 x 352+수평플립이 수행된다. 평가를 위해 비디오 프레임 간의 예측과 Ground truth사이의 PSNR과 SSIM점수, ground truth와 interpolated된 이미지 사이의 RMS차이로 정의되는 interpolation error를 보고한다.

**4.2.Ablation Studies(절제 연구?)**

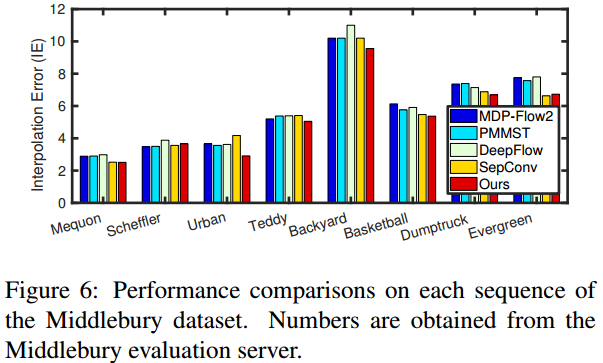
처음 두번의 실험에서는 무작위로 학습용으로 Adobe240fps 데이터셋에서 107개의 비디오와 테스트용으로 12개의 비디오를 추출했다.

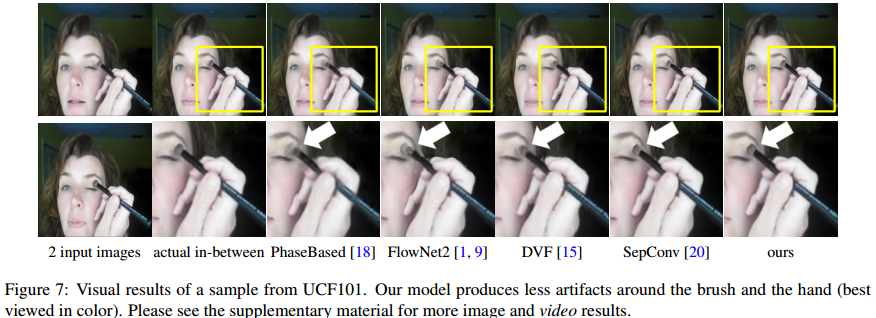
**Effectiveness of multi-frame video interpolation.** 우리는 먼저 프레임 사이에 여러 개를 공동으로 예측하는 것이 비디오 보간 결과를 개선하는지를 시험한다. 이를 위해 우리는 시간 단계에 고르게 분포되어 있는 중간의 단일, 3, 7프레임을 예측한다. 테스트할 때는 각 모델을 사용하여 7개의 중간 프레임을 예측한다. 표2는 학습 중에 중간 프레임을 더 많이 예측할수록 모델이 더 좋다는 것을 분명히 보여준다.

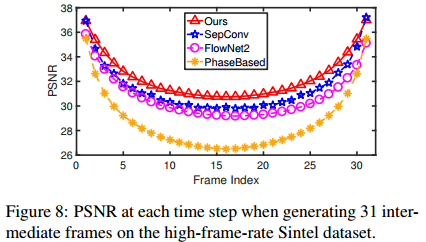
******Impact of different components design.** 우리는 각 요소의 기여도를 조사한다. 특히 두번째 U-Net에서 흐름 개선을 제거하여 흐름 interpolation의 영향을 연구한다(visibility maps은 계속 사용). 표3에서 각 세가지 구성 요소를 제거하면 성능에 해를 끼친다는 것을 확인할 수 있다. Visibility maps을 추가하면 interpolation 성능이 약간 개선된다. 없으면 그림3에서와 같이 모션 경계 주변에 생성된 아티팩트가 생성된다. Smoothness terms를 추가하면 성능이 약간 떨어지지만, 입력 이미지 간에 시각적으로 호소력 있는 optical flow를 생성하는 것이 유용하다고 생각한다.

**Impact of the number of training samples.** 마지막으로 우리는 학습 샘플의 개수의 효과를 조사한다. 우리는 Adobe240fps 데이터셋에 대해서만 학습된 모델과 전체 데이터셋에 대해 훈련된 모델, 이 두 모델을 비교한다. UCF101 데이터셋에서 이 두 모델의 성능은 표4의 마지막 두 행에서 확인할 수 있다. 우리는 우리의 모델이 더 많은 학습 자료로부터 이익을 얻는 것을 볼 수 있다.

**4.3. Comparison with state-of-the-art methods**

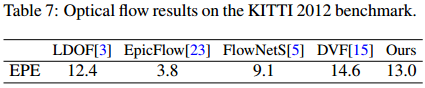
phase-based interpolation, separable adaptive convolution (SepConv), deep voxel flow (DVF)들과 우리의 접근방식을 비교한다. 또한 [1]에 제시된 interpolation 알고리즘을 사용해 baseline접근 방식을 구현하고, 여기서 FlowNet2를 사용하여 두 입력 이미지사이의 양방향 optical flow 결과를 계산한다. FlowNet2는 글로벌 배경 모션을 포착하고 optical flow에 대한 날카로운 모션 경계를 복구하는데 좋다. 따라서, occlusion reasoning와 결합할 때, FlowNet2는 강력한 baseline역할을 한다.

**Single-frame video interpolation.** SepConv외에도 interpolation 알고리즘의 optical flow method부분을 MDP-Flow2, PMMST, DeepFlow로 각자 결합해서 비교했다. 우리 모델은 8개 시퀀스중 6개의 항목에서 최고의 성능을 발휘한다. 특히 Urban시퀀스는 합성적으로 생성되며, Teddy시퀀스는 실제로 2개의 스테레오 페어를 포함한다. 우리 모델의 성과는 우리의 접근법의 일반화 능력을 입증한다.

**Multi-frame video interpolation**. slowflow데이터셋의 경우, 7개의 중간 프레임을 예측한다. 모든 실험은 공간 크기가 1280x1024인 half-resolution이미지에 대해 수행된다. 이 데이터셋에서 우리의 접근법은 최고의 PSNR, SSIM점수를 달성하고 FlowNet2는 최고의 SSIM, L1 error점수를 달성한다. FlowNet2는 글로벌 모션 캡쳐에 능하여 글로벌 모션 패턴을 따르는 배경 영역에 대한 예리한 예측 결과를 생성한다. 높은 프레임률의 Sintel 데이터셋에서 우리의 접근법은 다른 모든 방법을 크게 능가한다. 우리의 접근법은 마지막 단계의 SepConv보다 약간 더 나쁜 것을 제외하고 모든 데이터셋에 대해 최고의 결과를 달성하여 단일또는 여러 개의 중간 프레임을 생성한다. 우리의 모델이 수정 없이 다른 시나리오에 직접 적용될 수 있다는 사실을 고려하면 주목할 만하다.

**4.4. Unsupervised Optical Flow**

우리의 비디오 프레임 interpolation 접근법은 두 입력이미지 사이의 양방향 optical flow를 계산할 수 있는 비지도 네트워크(the flow computation CNN)를 가지고 있다. 서로 다른 방법의 end-point error(EPE)점수는 표7에 나와있다. 이전의 비지도 방법인 DVF와 비교했을 때, 우리 모델은 평균 EPE 13.0으로, 11%의 상대적 개선을 달성했다. DVF가 우리와 유사한 U-Net 아키텍쳐를 가지고 있기 때문에 멀티 프레임 비디오 interpolation 설정에서 이러한 개선 결과가 발생할 가능성이 매우 높다.



**5. Conclusion**

두개의 입력이미지 사이에 필요한 만큼의 중간 비디오 프레임을 생성할 수 있는 end-to-end로 학습할 수 있는 CNN을 제안했다. Flow computation CNN을 사용하여 두 프레임 사이의 양방향 optical flow를 추정하고, 두 흐름 필드를 선형적으로 융합하여 중간 optical flow 필드를 근사치에 가깝게 한다. 그런 다음 흐름 interpolation CNN을 사용하여 근사치 흐름 필드를 개선하고 interpolation을 위한 soft visibility maps을 예측한다. 우리는 네트워크가 7개의 중간 프레임을 예측하도로고 학습시키기 위해 1.1K 240fps이상의 비디오 클립을 사용한다. 별도의 검증셋에 대한 Ablation 연구는 flow interpolation 및 visibility maps의 이점을 입증한다. 우리의 멀티프레임 접근 방식은 Middlebury, UCF101, slowflow, high-frame-rate Sintel 데이터셋에서 최신 단일 프레임 방싯보다 지속적으로 더 우수한 성능을 발휘한다. Optical flow에 대한 비지도학습으로 인해 우리의 네트워크는 KITTI 2012 벤치마크에서 최근의 DVF방법을 능가한다.