2024 겨울 세미나

Hand Pose Estimation

Sogang University
Vision & Display Systems Lab, Dept. of Electronic Engineering

Presented By
MinSuh Song

Outline

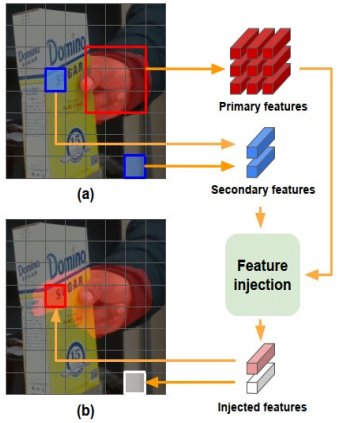
- JoonKyu Park, Yeonguk Oh, et al. **"HandOccNet: Occlusion-Robust 3D Hand Mesh Estimation Network."** CVPR, 2022

- Qichen Fu, Xingyu Liu, et al. **"Deformer: Dynamic Fusion Transformer for Robust Hand Pose Estimation."** ICCV, 2023

HandOccNet: Occlusion-Robust 3D Hand Mesh Estimation Network

HandOccNet

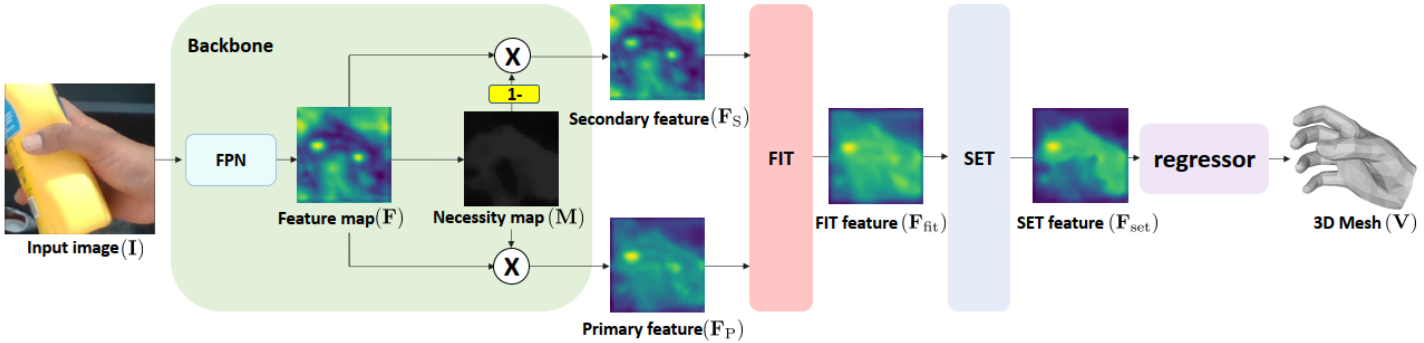
- Abstract
  - 손이 가려진 부분의 정보를 이용하여 3D hand mesh를 재구성하고자 함
  - HandOccNet은 두 개의 Transformer 기반 모델로 구성됨
    - Feature Injecting Transformer (FIT)
      - 손과 연관 있는 occluded region에 hand information을 주입(inject)
    - Self-enhancing Transformer (SET)
      - Self-attention을 이용하여 FIT의 결과를 refine
  - HO-3D, FPHA Dataset에 대한 mean joint error, F-scores에서 SOTA 달성



<HandOccNet 예시>

HandOccNet

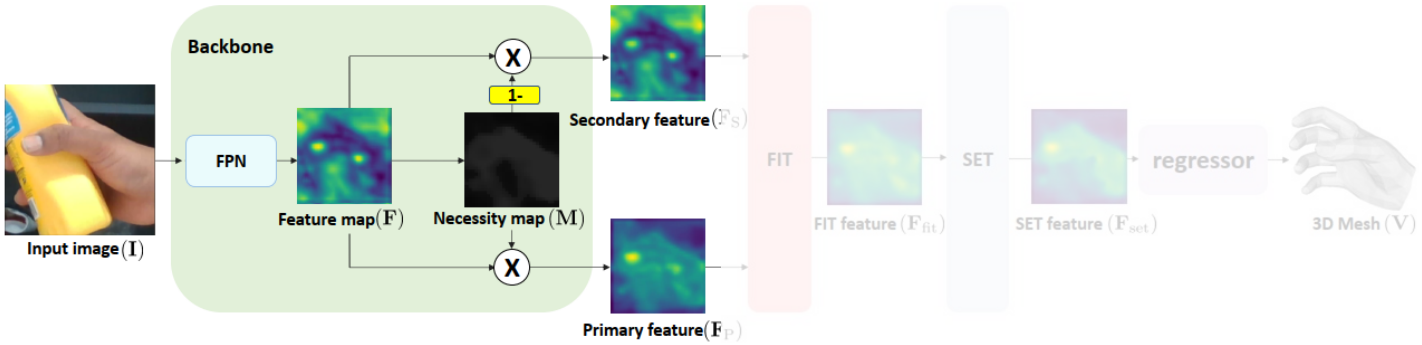
- Introduction
  - 기존의 연구들은 spatial attention mechansim을 사용
  - Spatial attention map을 생성해서 network가 어디에 focus해야 하는지 알려줌
    - Occluded region의 비중을 낮춰서 human region에 focus할 수 있도록 guide
  - Limitations
    - 3D hand coordinate를 얻기에 적합하지 않음
    - 손이 occluded된 부분이 많으면, hand region에 대한 정보가 부족
  - HandOccNet은 occluded region의 정보를 이용하여 hand region을 보완하고자 함



<HandOccNet의 전체적인 구조>

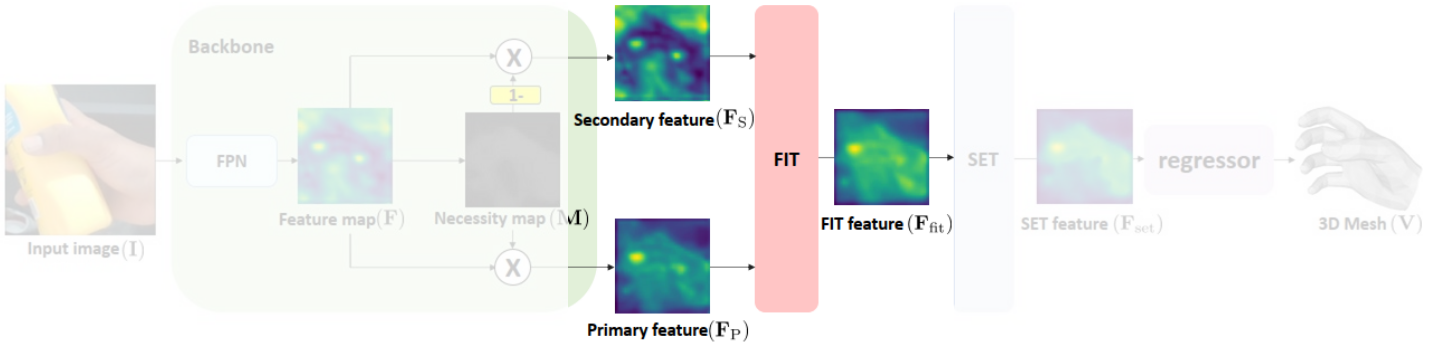
HandOccNet

- Backbone
  - 이미지를 입력 받아 feature map(F) 생성, F를 이용하여 necessity map(M) 생성
    - Necessity map은 이미지에서 어떤 픽셀에 focus해야 하는지 알려주는 spatial importance를 제공
  - Primary feature(F_p)
    - Primary feature = hand region
    - $F_p = F \otimes M$
  - Secondary feature(F_s)
    - Secondary feature = occluded region
    - $F_s = F \otimes (1 - M)$



HandOccNet

- FIT
  - FIT은 F_p와 F_s를 입력 받아, F_p와 연관이 있다고 판단되는 F_s에 F_p의 정보를 inject
    - Occlusion을 발생시키는 object information은 손과 매우 연관이 있을 수 있기에 F_s는 어디에 F_p를 inject해야 하는지 알려주는 역할을 함
    - 기존의 연구는 F_p에만 집중하고, F_s의 값은 최대한 억제
    - HandOccNet은 부족한 F_p의 정보를 F_s를 이용하여 보완



HandOccNet

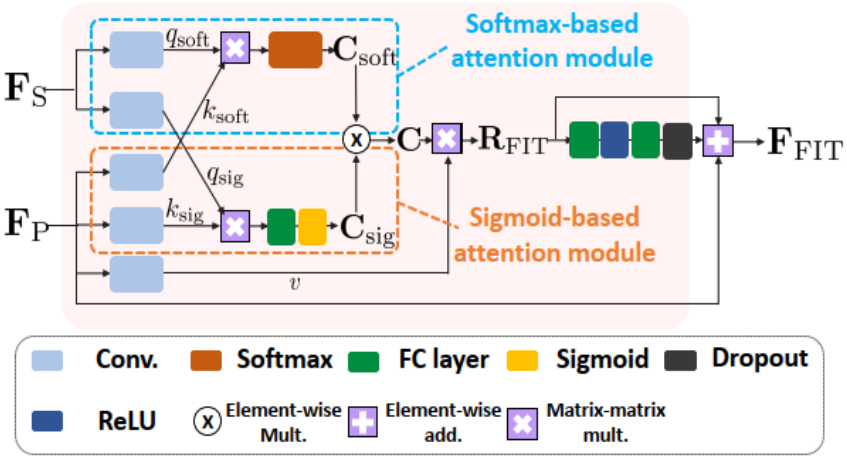
- FIT
  - Softmax based attention module

$$-C_{soft} = softmax(\frac{q_{soft}k_{soft}^{T}}{\sqrt{d_{k_{soft}}}})$$

query q_{soft}는 F_s로부터 추출, key k_{soft}는 F_p로부터 추출

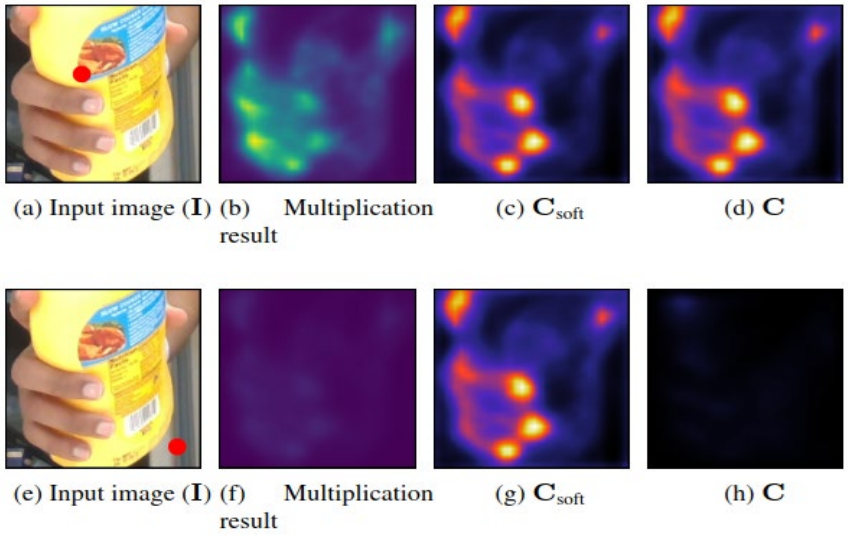
Correlation map C_{soft}는 각 픽셀에서 q_{soft}와 k_{soft}의 연관된 정도를 나타내는 역할

C_{soft}를 이용하여 F_s의 자리에 inject하기 적합한 F_p를 찾음



HandOccNet

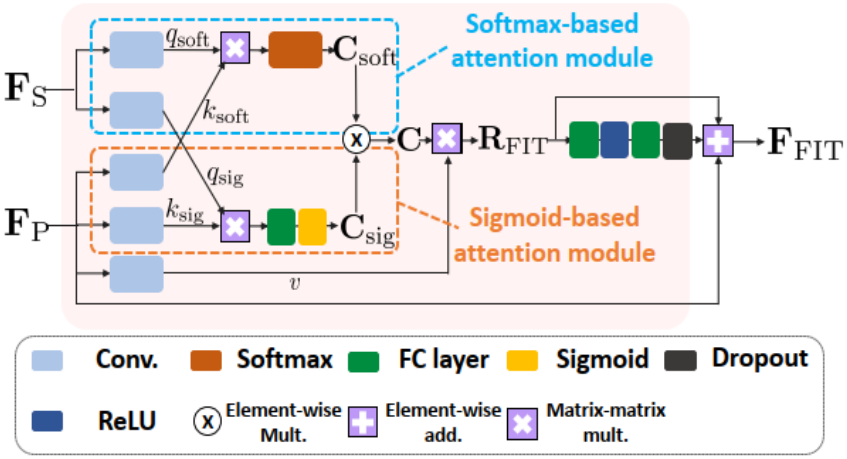
- FIT
  - Softmax based attention module
    - Softmax based attention module의 한계점
      - 특정 query pixel과 모든 key들이 연관이 없다고 판단될 때, undesired high correlation score 발생
    - Undesired high correlation score



<Undesired high correlation score>

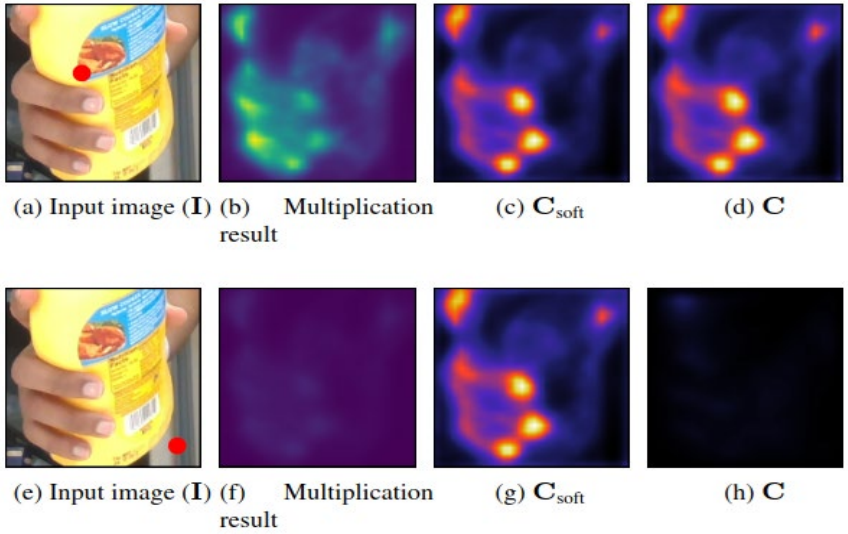
HandOccNet

- FIT
  - Sigmoid based attention module
    - $C_{sig} = sigmoid(pool\left(\frac{q_{sig}k_{sig}^{T}}{\sqrt{d_{k_{sig}}}}\right))$
    - SoftMax 함수는 입력 element를 다른 element들과 관계를 고려하여 확률 분포로 정규화
    - Sigmoid 함수는 입력 element만을 확률로 정규화
      - Query와 key의 유사도가 작으면 주위에 상관없이 작은 attention-score를 부여



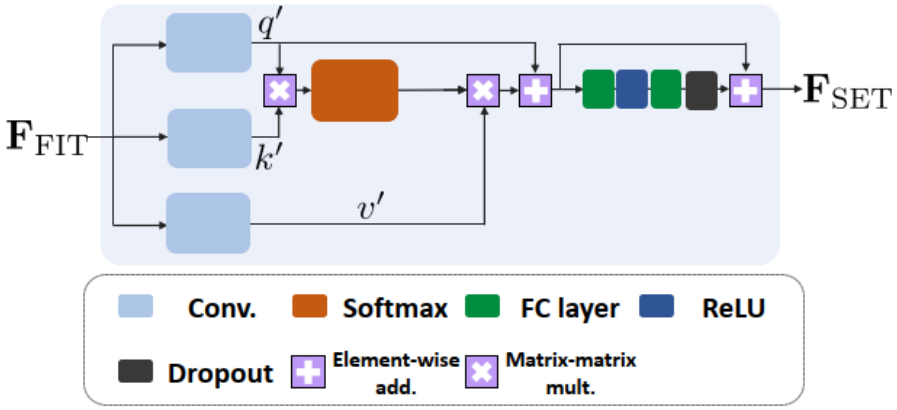
HandOccNet

- FIT
  - Sigmoid based attention module
    - $C = C_{soft} \otimes C_{sig}$
    - Undesired high correlation score 문제 해결



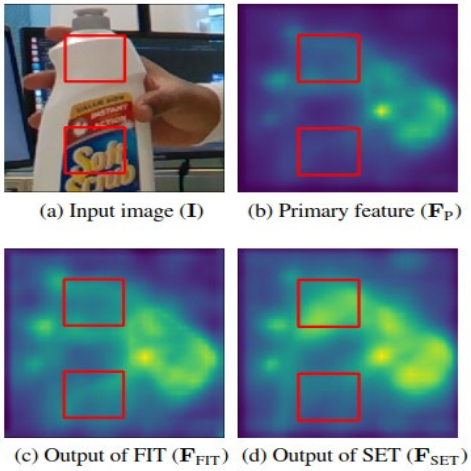
HandOccNet

- SET
  - F_{FIT} : correlation map C에 따라 hand information을 occluded region에 주입한 결과
  - SET은 distant information을 바탕으로 self-attention을 통해 F_{FIT}을 refine
  - SoftMax의 역할
    - Self-attention의 query와 key의 유사도를 알려주는 correlation map을 생성하기 위함
  - F_{FIT}은 SET를 통과하면서 F_{SET}로 refine



HandOccNet

- SET
  - Input image 그림(a)로부터 hand information을 담고 있는 Primary feature (F_p) 그림(b) 추출
  - 그림(c): FIT을 통해 occluded region에 연관된 F_p를 inject
  - 그림(d): SET을 이용해 FIT의 결과를 self-enhancing
    - 그림(c)의 손가락 끝과 손바닥이 연결된 부분
    - 그림(c)의 손과 연관이 없는 occluded region의 비중 감소



HandOccNet

- Evaluating Hand Pose Estimation Models
  - HO-3D와 FPHA dataset를 기준으로 mean joint error, mesh error, F-scores로 성능 측정
    - Mean joint error: 추정된 joint와 실제 joint의 위치 차이의 평균
    - Mesh error in mm: 추정된 hand mesh와 실제 hand mesh의 거리 차이의 평균
    - F-scores: 추정된 hand mesh와 실제 hand mesh 사이의 일치하는 정도
      - F@5: 5mm의 오차를 허용했을 때, 측정값과 실제 값 사이의 일치하는 정도
      - F@15: 15mm의 오차를 허용했을 때, 측정값과 실제 값 사이의 일치하는 정도

Methods	Joint	Mesh	F@5	F@15
Pose2Mesh [6]	12.5	12.7	44.1	90.9
Hasson *et al.* [14]	11.4	11.4	42.8	93.2
I2L-MeshNet [26]	11.2	13.9	40.9	93.2
Hasson *et al.* [15]	11.1	11.0	46.0	93.0
Hampali *et al.* [13]	10.7	10.6	50.6	94.2
METRO [21]	10.4	11.1	48.4	94.6
Liu *et al.* [23]	10.2	9.8	52.9	95.0
HandOccNet (Ours)	**9.1**	**8.8**	**56.4**	**96.3**

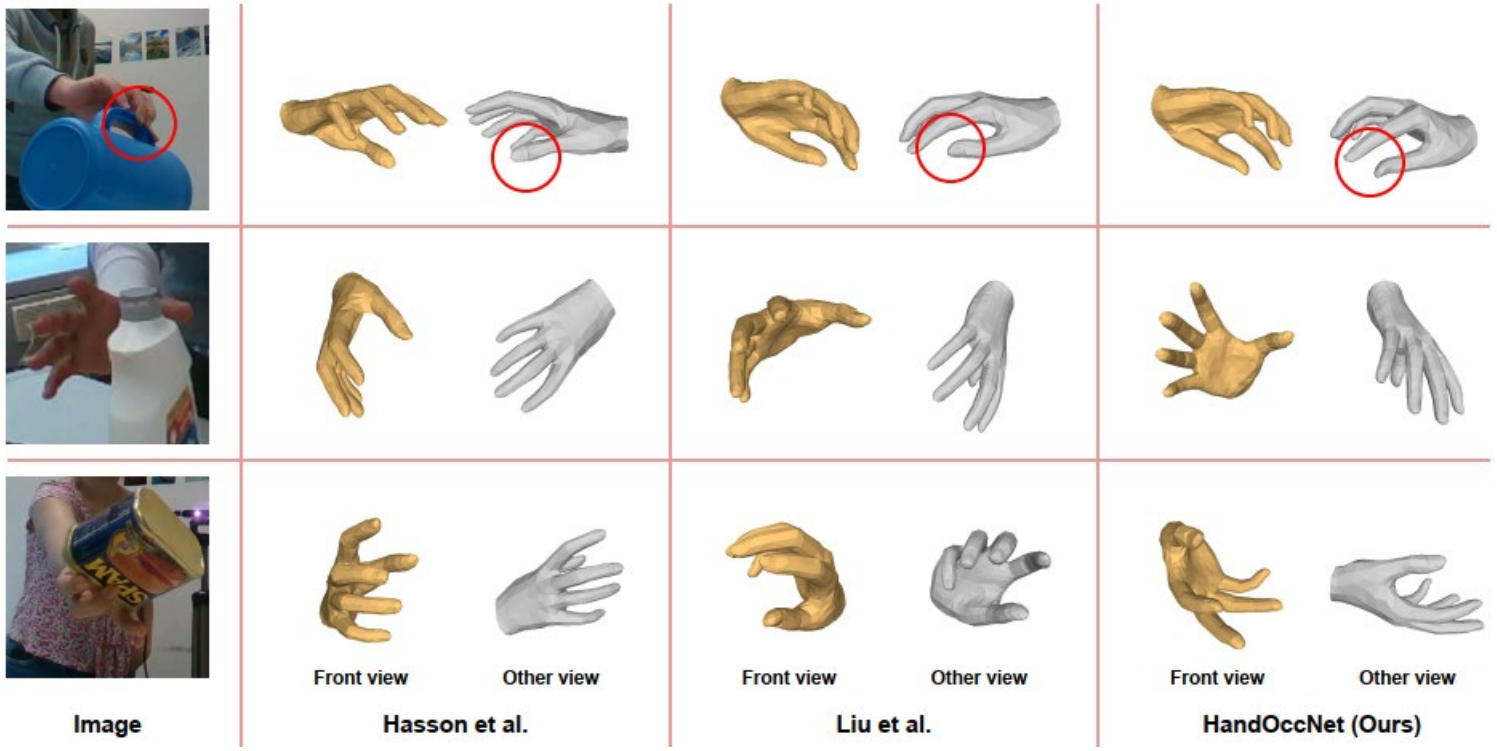
< HO-3D에서 HandOccNet 성능비교표>

Methods	3D joint error
I2L-MeshNet [26]	21.2
Hasson *et al.* [14]	18.0
Liu *et al.* [23]	16.0
Hasson *et al.* [15]	14.9
HandOccNet (Ours)	**10.8**

< FPHA에서 HandOccNet 성능비교표>

HandOccNet

- Experiment



< 기존 모델과 HandOccNet의 정성적 평가>

HandOccNet

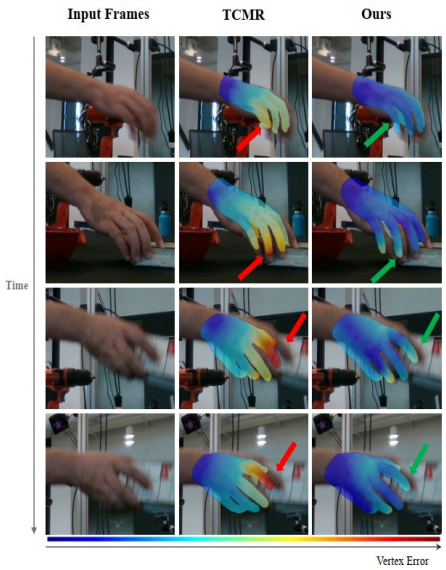
- Experiment

< 기존 모델과 HandOccNet의 정성적 평가>

Deformer: Dynamic Fusion Transformer for Robust Hand Pose Estimation

Deformer

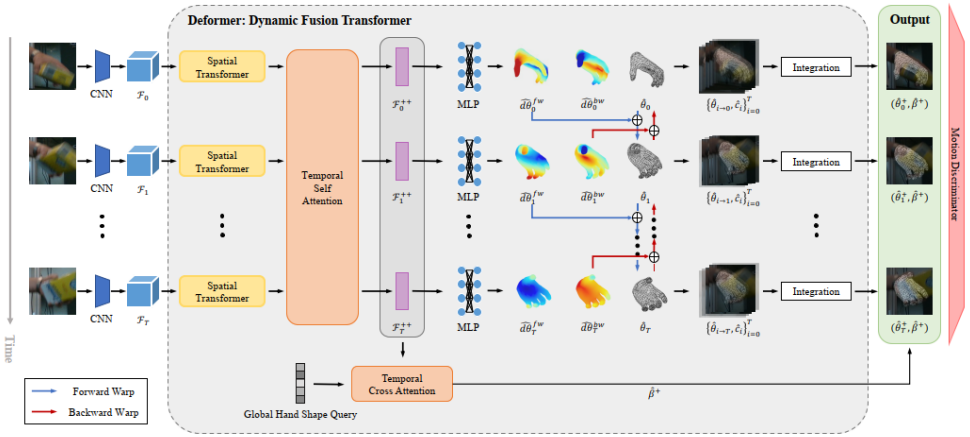
- Abstract
  - Video (sequence of frames)에서 3D hand pose estimation을 수행
  - 현재 image frame에서 occlusion이나 blur 현상이 발생하면, 전 후 frame의 hand information을 이용해서 estimation을 수행
  - Dynamic Fusion Module
    - 다른 시간대의 정보를 이용해서 현재 프레임의 hand pose를 deform
  - maxMSE Loss
    - 새로운 loss function 제시



<Deformer 예시>

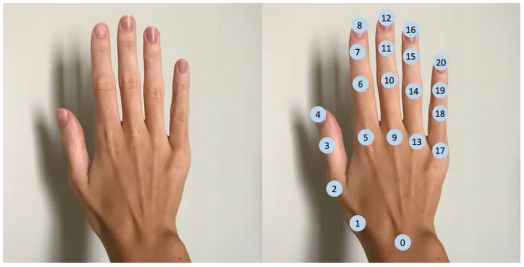
Deformer

- Introduction
  - Deformer는 spatial transformer, temporal transformer, dynamic fusion module로 구성
    - 영상이 입력되면, CNN을 이용하여 프레임 마다 hand feature를 추출
    - Spatial transformer
      - 한 프레임에서 손을 나타내는 픽셀들의 관계를 파악
    - Global temporal transformer
      - 각 timestamp 마다 프레임 사이의 관계를 파악
    - Dynamic fusion model
      - 전후 프레임의 정보를 이용하여 현재 프레임의 추정된 hand pose를 deform

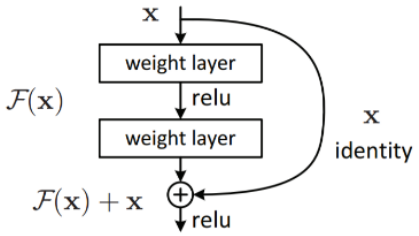


Deformer

- Spatial transformer
  - 이미지의 전체적인 구조를 파악하면서 hand information에 focus 하기 위해 spatial transformer를 활용
    - Video가 입력되면 CNN을 이용하여 매 순간의 timestamp(t)마다 initial hand feature($\mathcal{F}_t$)를 추출
    - $\mathcal{F}_t$의 sequence를 입력 받아 enhanced 된 feature $\mathcal{F}_t^e$를 생성
      - $\mathcal{F}_t^e$는 hand feature vector들의 non-local interaction에 대한 정보를 포함
    - $\mathcal{F}_t^e$로부터 N_j=21 joints location prediction $\hat{\mathcal{J}}_t^{2D}$를 추출
    - Skip connection의 idea를 활용, decoder에서 cross attention을 통해 $\mathcal{F}_t^e$와 $\hat{\mathcal{J}}_t^{2D}$를 결합
  - 결과적으로 하나의 image에서 손의 모양을 추정하고 joint를 찾음 ($\mathcal{F}_t^+$)



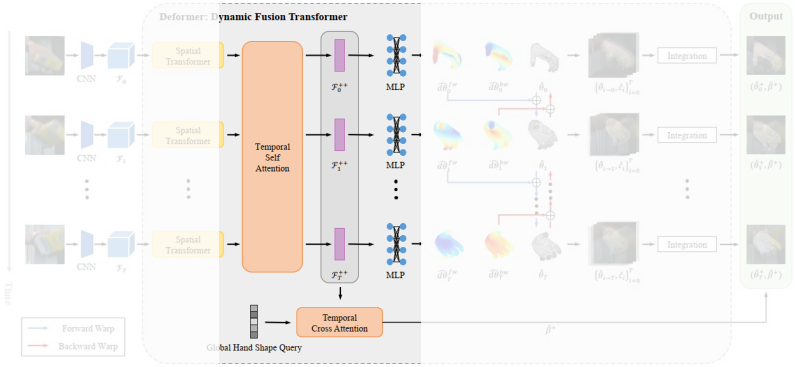
< Hand keypoints 21 joints>



< Skip connection>

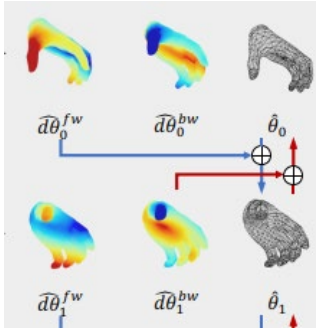
Deformer

- Temporal Transformer
  - 현재 프레임에서 occlusion이나 blur가 발생하면, 전 후 시점의 프레임을 참고하여 estimation 수행
    - Transformer의 self-attention으로 인해 프레임들끼리 시간적 거리와 관계없이 정보를 주고 받을 수 있음
    - $\{\mathcal{F}_t^+\}_{t=1}^T$를 입력 받아서 새로운 latent vector $\{\mathcal{F}_t^{++}\}_{t=1}^T$를 출력
    - $\{\mathcal{F}_t^{++}\}_{t=1}^T$를 MLP에 전달하여 MANO pose parameter $\hat{\theta}_t$를 생성
    - $\hat{\theta}_t$만을 이용하여 estimation을 수행하면 전체적인 손의 모양의 일관성이 유지되지 않음
      - Global hand shape query와 $\{\mathcal{F}_t^{++}\}_{t=1}^T$를 cross-attention하여 global shape parameter $\hat{\beta}^+$를 예측



Deformer

- Dynamic fusion module
  - Temporal transformer의 결과인 $\{\mathcal{F}_t^{++}\}_{t=1}^{T}$만을 사용하면 현재 프레임에 대한 dependency가 강함
  - 전후 프레임의 정보를 단순히 참고하는 것이 아닌, 이를 이용하여 현재 프레임의 추정된 hand pose를 deform
  - $\{\mathcal{F}_t^{++}\}_{t=1}^{T}$에서 추가로 confidence score $\hat{c}_t$와 현재 모션으로부터 전후 시점의 hand motion $(\hat{d}\theta_t^{fw}, \hat{d}\theta_t^{bw})$를 추출
    - Forward motion: $\hat{\boldsymbol{d}}\boldsymbol{\theta}_{\boldsymbol{t}}^{\boldsymbol{fw}} = \theta_{t+1} - \theta_t$ (다음 시점의 MANO parameter와 현재 값과의 차이)
    - Backward motion: $\hat{\boldsymbol{d}}\boldsymbol{\theta}_{\boldsymbol{t}}^{\boldsymbol{bw}} = \theta_{t-1} - \theta_t$ (이전 시점의 MANO parameter와 현재 값과의 차이)



<Dynamic Fusion Module>

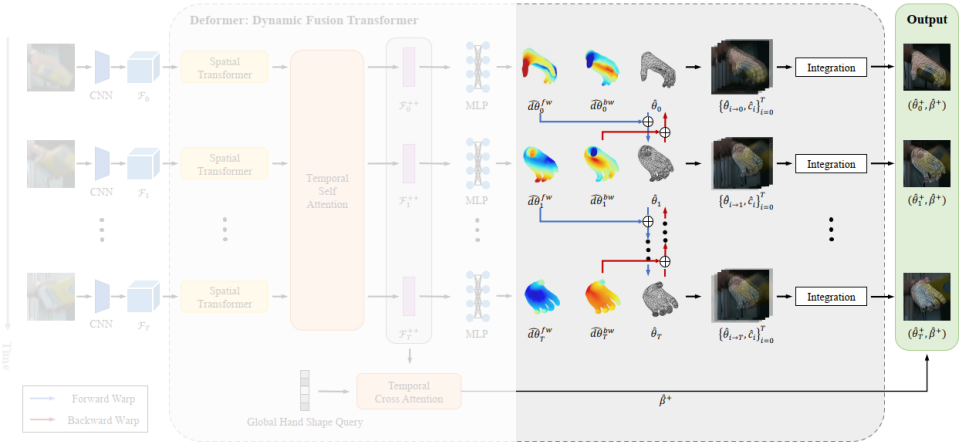
Deformer

- Dynamic fusion module
  - Forward, backward motion을 이용하면 frame j의 hand pose parameter를 frame i가 영향을 주어 변형할 수 있음

$$\hat{\theta}_{i\to j} = \begin{cases} \hat{\theta}_i + \sum_{k=j}^{i-1} \hat{d}\theta_k^{bw} & (if\ j < i) \\ \hat{\theta}_i + \sum_{k=i}^{j-1} \hat{d}\theta_k^{fw} & (if\ j > i) \end{cases}$$

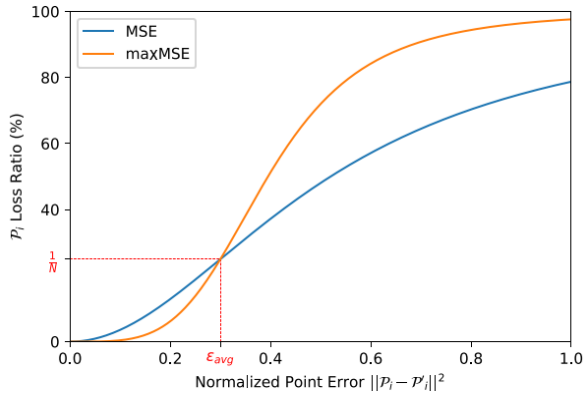
  - 모든 timestamp t의 변형된 hand pose를 합성하여 최종 hand pose parameter $\hat{\theta}_t^+$를 생성함

$$\hat{\theta}_t^+ = \frac{\sum_{t'=1}^{T} e^{\hat{c}_{t'}} \hat{\theta}_{t'\to t}}{\sum_{t'=1}^{T} e^{\hat{c}_{t'}}}$$

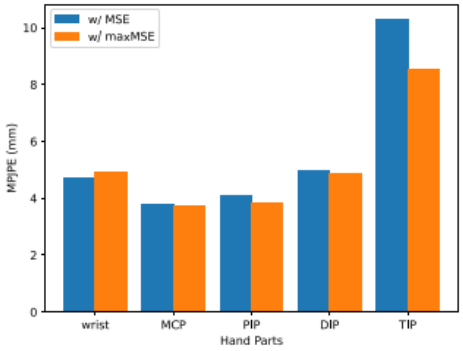


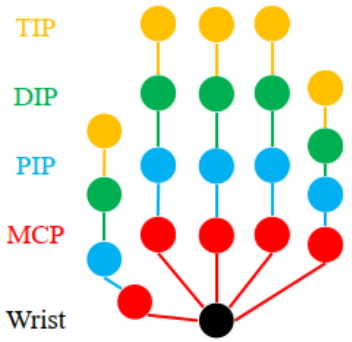
Deformer

- maxMSE Loss
  - 기존의 standard MSE loss는 손의 서로 다른 부분들에 대해서 고르지 않은 에러를 보임
    - MSE loss를 사용한 모델은 손바닥에서 특히 낮은 에러와, 손가락과 같은 섬세하고 복잡한 부분에 대해서는 높은 에러를 보임
  - 본 논문은 새로운 loss function으로 maxMSE를 제안
  - maxMSE loss는 large error를 가진 joint나 vertices에는 large weight를 부여하고, well-predicted parts에는 lower weight를 부여함



<Error에 따라 부여되는 weight 비교>





<Error imbalance issue가 완화된 모습>

Deformer

- Motion Discriminator
  - Temporal Incontinuity
    - 영상을 프레임 단위의 이미지로 나누고, 이를 다시 합쳐서 재생했을 때 매끄럽지 못하고 툭툭 끊기는 현상
  - Motion discriminator는 hand mesh sequence를 입력 받아 적합성을 판별
    - Deformer를 generator로 하고 motion discriminator를 이용하여 adversarial training을 진행

maxMSE	Motion Discrimination	All	Occlusion (25%-50%)	Occlusion (50%-75%)	Occlusion (75%-100%)
✗	✗	5.72 (88.6)	6.11 (87.8)	6.14 (87.7)	6.75 (86.5)
✗	✓	5.63 (88.7)	6.00 (88.0)	5.92 (88.2)	6.62 (86.8)
✓	✗	5.43 (89.1)	6.05 (87.8)	6.00 (88.0)	6.59 (86.8)
✓	✓	**5.22 (89.6)**	**5.71 (88.6)**	**5.70 (88.6)**	**6.34 (87.3)**

<Ablation Study>

Deformer

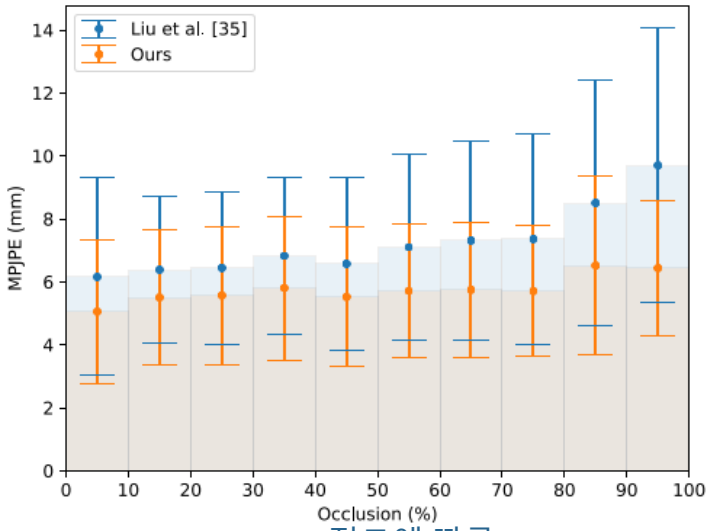
- Experiment
  - HO-3D와 DexYCB dataset를 기준으로 mean per joint position error, F-scores, AUC scores로 성능 측정
    - Mean per joint position error (MPJPE): 추정된 joint와 실제 joint의 위치 차이의 평균
    - F-scores: 추정된 hand mesh와 실제 hand mesh 사이의 일치하는 정도
      - F@5: 5mm의 오차를 허용했을 때, 측정값과 실제 값 사이의 일치하는 정도
      - F@15: 15mm의 오차를 허용했을 때, 측정값과 실제 값 사이의 일치하는 정도
    - AUC scores: 모델이 성공적으로 결과를 판단하는 정도

Methods	Input	All	Occlusion (25%-50%)	Occlusion (50%-75%)	Occlusion (75%-100%)
A2J [55]	Depth	12.07 (76.0)	12.44 (75.3)	14.74 (70.7)	19.59 (61.5)
[49] + ResNet50	Monocular	7.12 (85.8)	7.65 (84.7)	8.73 (82.6)	11.90 (76.3)
[49] + HRNet32	Monocular	6.83 (86.4)	7.22 (85.6)	8.00 (84.0)	10.65 (78.8)
MeshGraphormer [33]	Monocular	6.41 (87.2)	6.85 (86.3)	7.22 (85.6)	7.76 (84.5)
[36]	Monocular	6.33 (87.4)	6.70 (86.6)	7.17 (85.7)	8.96 (82.1)
[40]	Monocular	**5.80** (**88.4**)	**6.22** (**87.6**)	**6.43** (**87.2**)	**7.37** (**85.3**)
$S^2HAND(V)$ [51]	Sequence	7.27 (85.5)	7.74 (84.5)	7.71 (84.6)	7.87 (84.3)
VIBE [29]	Sequence	6.43 (87.1)	6.72 (86.5)	6.84 (86.4)	7.06 (85.8)
TCMR [10]	Sequence	6.28 (87.5)	6.56 (86.9)	6.58 (86.8)	6.95 (86.1)
Ours	Sequence	**5.22** (**89.6**)	**5.71** (**88.6**)	**5.70** (**88.6**)	**6.34** (**87.3**)

<DexYCB에서 MPJPE (AUC) 비교표>

Deformer

- Experiment



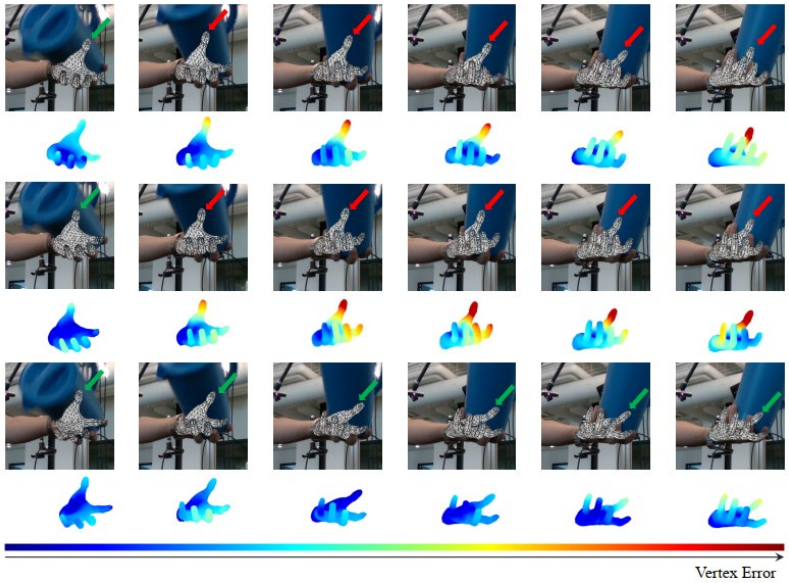
<Occlusion 정도에 따른 MPJPE>

Methods	Input	Hand Error (↓)		Hand F-score (↑)	
		Joint	Mesh	F@5	F@15
[20]	Monocular	11.1	11.0	46.0	93.0
[17]	Monocular	10.7	10.6	50.6	94.2
[36]	Monocular	10.1	9.7	53.2	95.2
[40]	Monocular	**9.1**	**8.8**	**56.4**	**96.3**
VIBE [29]	Sequence	9.9	9.5	52.6	95.5
TCMR [10]	Sequence	11.4	10.9	46.3	93.3
TempCLR [61]	Sequence	10.6	10.6	48.1	93.7
Ours	Sequence	**9.4**	**9.1**	**54.6**	**96.3**

<MPJPE, F-scores 비교표>

Deformer

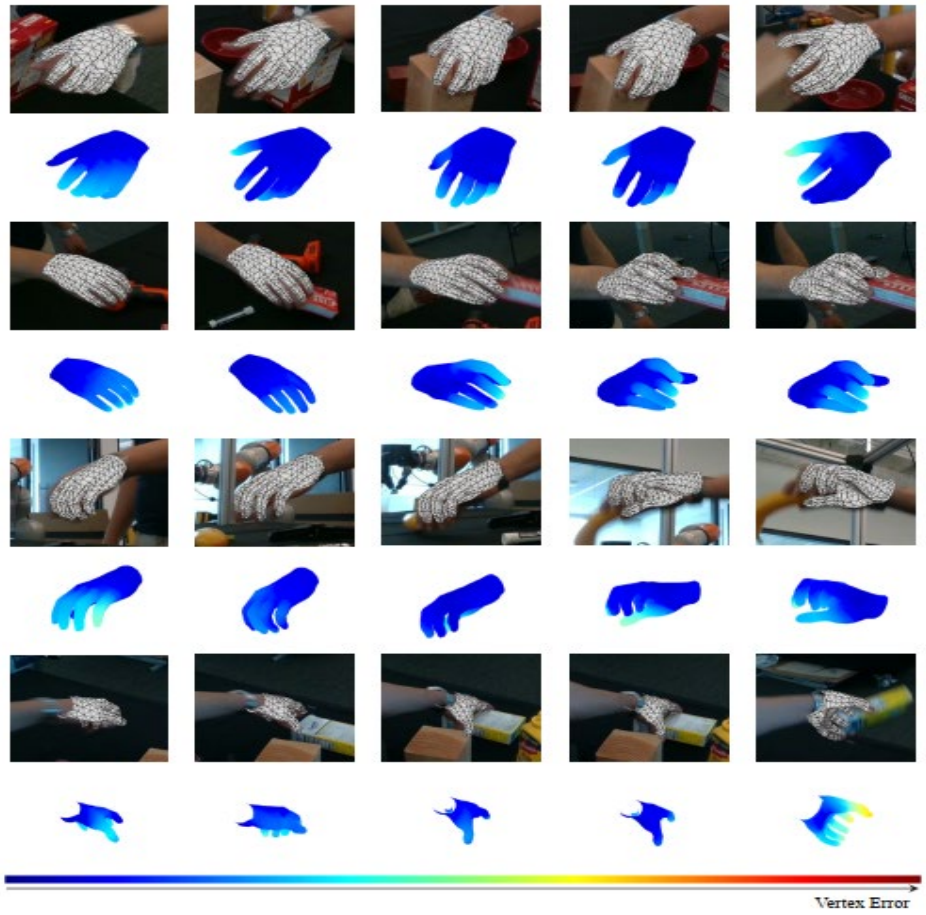
- Experiment
  - (Top): Semi-Hand-object
    - Single-view method
  - (Middle): Mask R-CNN
    - Video-based method
  - (Bottom): Deformer



<기존 모델들과 Deformer의 정성적 평가>

Deformer

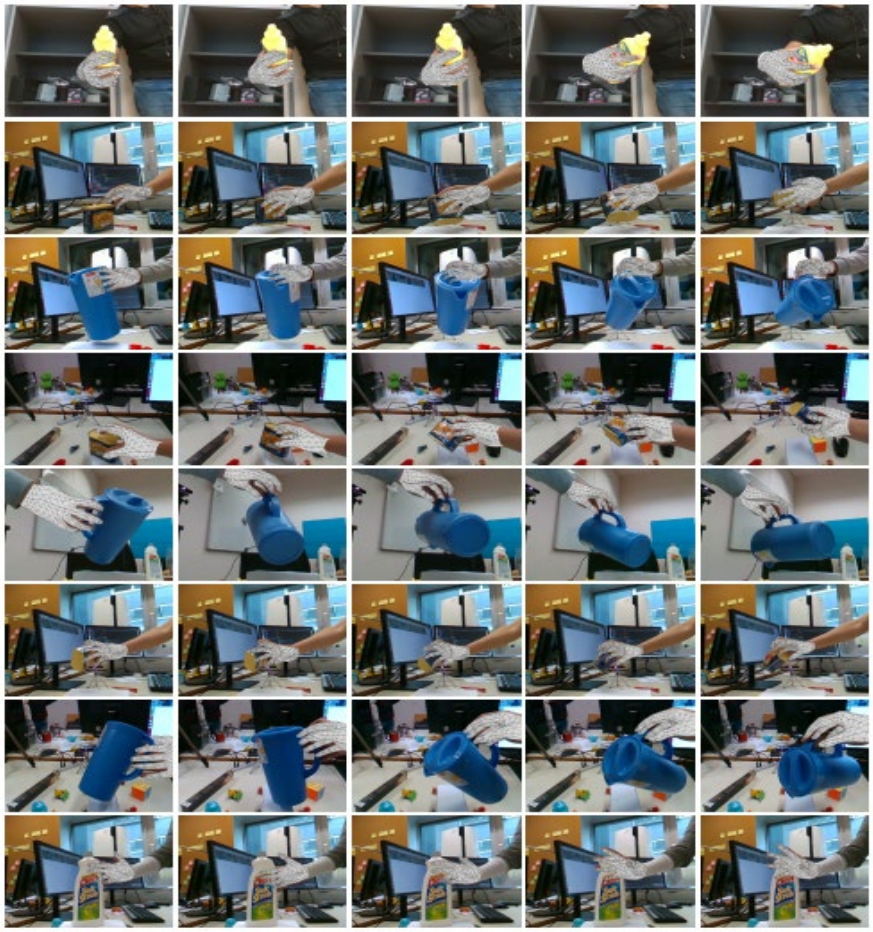
- Experiment



<DexYCB dataset을 이용한 정성적 결과>

Deformer

- Experiment



<HO-3D dataset을 이용한 정성적 결과>

감사합니다