Unsupervised Human Pose Estimation

양 창 희

Vision & Display Systems Lab.

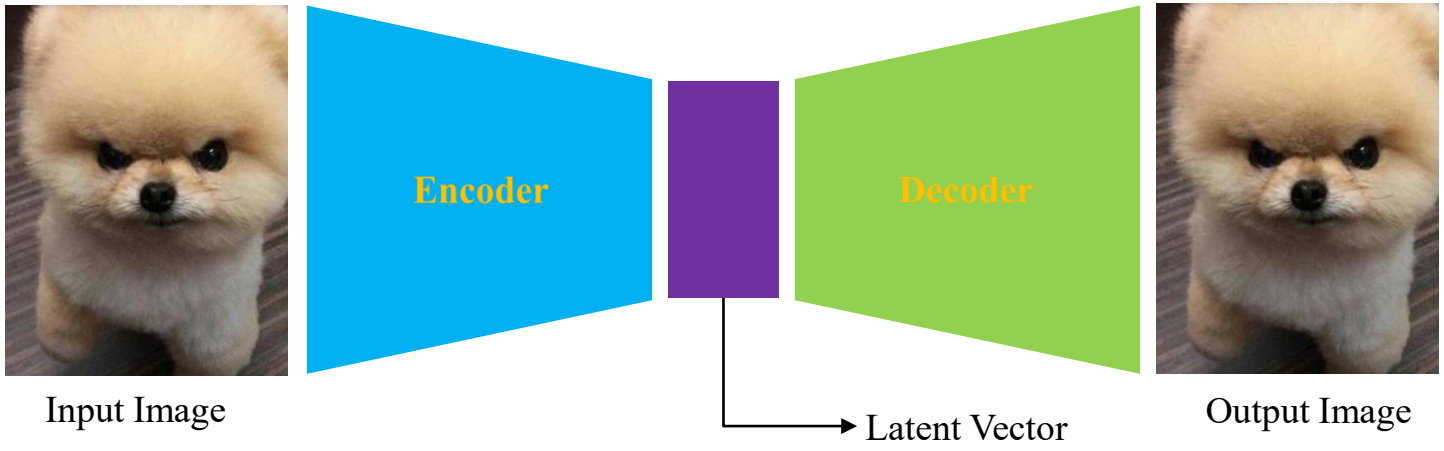
Dept. of Electronic Engineering, Sogang University

Outline

- Background
  - Auto-encoder
  - CycleGAN
- Development
  - Auto-encoder 관련 3d human pose estimation 논문
    - Unsupervised Geometry-Aware Representation for 3D human Pose Estimation – ECCV 2018
  - CycleGAN 관련 3d human pose estimation 논문
    - Unsupervised 3D Pose Estimation With Geometric self-supervision – CVPR 2019
  - **New Idea**
    - Unsupervised Human Pose Estimation Through Transforming Shape Templates – CVPR 2021
- Reference

Background

- Auto-Encoder

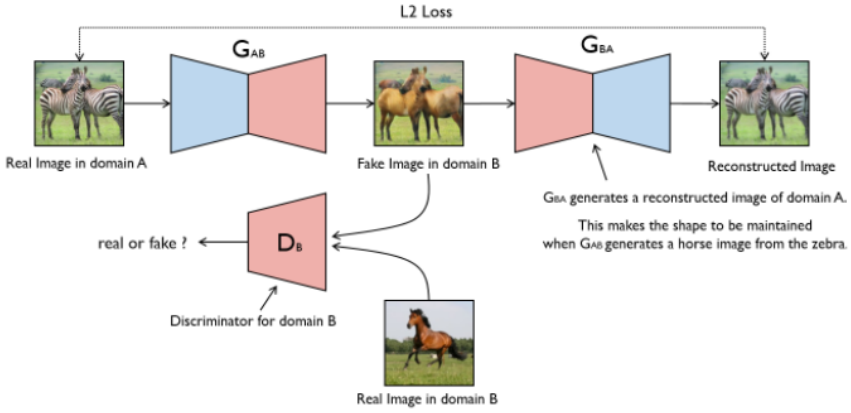


▪ Latent Vector => Input Image가 Encoder에 들어가면서 Latent Vector로 변함

- Compression 개념으로 Input Image의 중요한 정보를 Latent Vector에 넣음

Background

- Cycle Gan



- Input Image를 Cycle Gan에 넣으면 Output 또한 Input Image가 됨
  - L2 Loss를 이용하여 Input Image 자신을 훈련 시킴

- Discriminator는 Real or Fake만을 판단
  - Fake Image in Domain B와 Real Image in Domain B를 판단, 훈련 함

- Cycle Gan에서 Idea를 얻어 Self-Supervised Learing이 생김

Development

- Based on Auto-Encoder

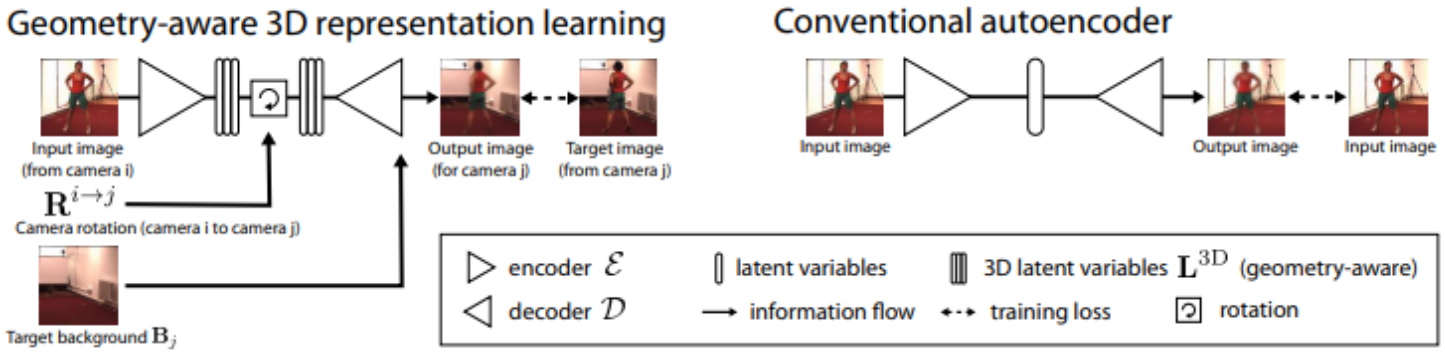
  - Unsupervised Geometry-Aware Representation for 3D human Pose Estimation – ECCV 2018

    - Contribution

      - Rotation matrix를 이용해 3d latent variable을 바꾸어 pose 정보를 얻어냄
      - 매우 적은 GT를 이용해 3D pose estimation을 성능을 향상 시킴

Development

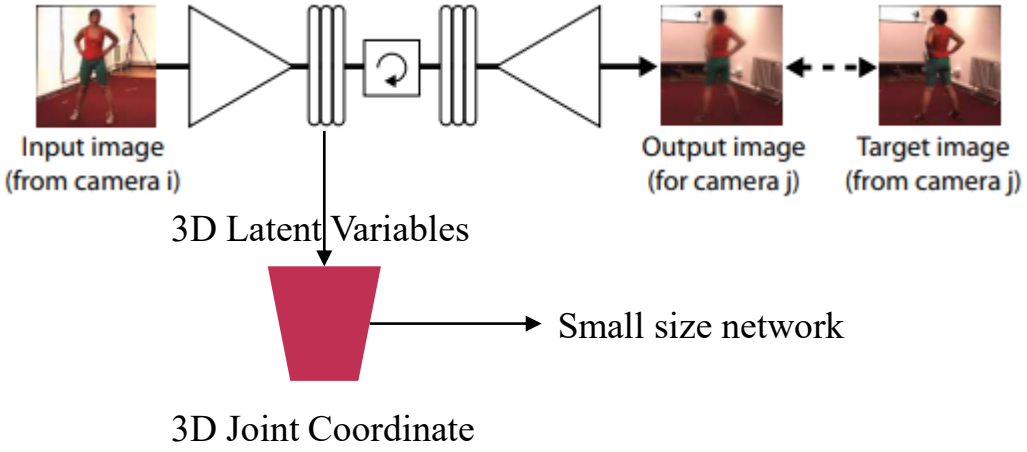
- Based on Auto-Encoder
  - Geometry-Aware 3D representation learning



- 3D Latent Variables를 만들어 Cam_i, Cam_j의 Rotation Matrix를 적용
  - 3D Latent Variables에는 Pose에 대한 정보가 있다고 추정
- Decoder 단에 Background Image를 넣어 Cam_j의 image와 loss를 구함
  - Background Image를 넣는 이유는 Pose에 대한 정보만이 3D Latent Variables에 들어가길 바람

Development

- Based on Auto-Encoder
  - 3D Human Pose Estimation
    - 3D Latent Variables에 Pose의 정보가 들어 있기 때문에 간단한 Network를 통과 시킴
      - 여기서 GT가 사용 됨
      - 저자들은 약 0.1~1%의 GT만을 사용 했다고 주장



Development

- Based on CycleGAN

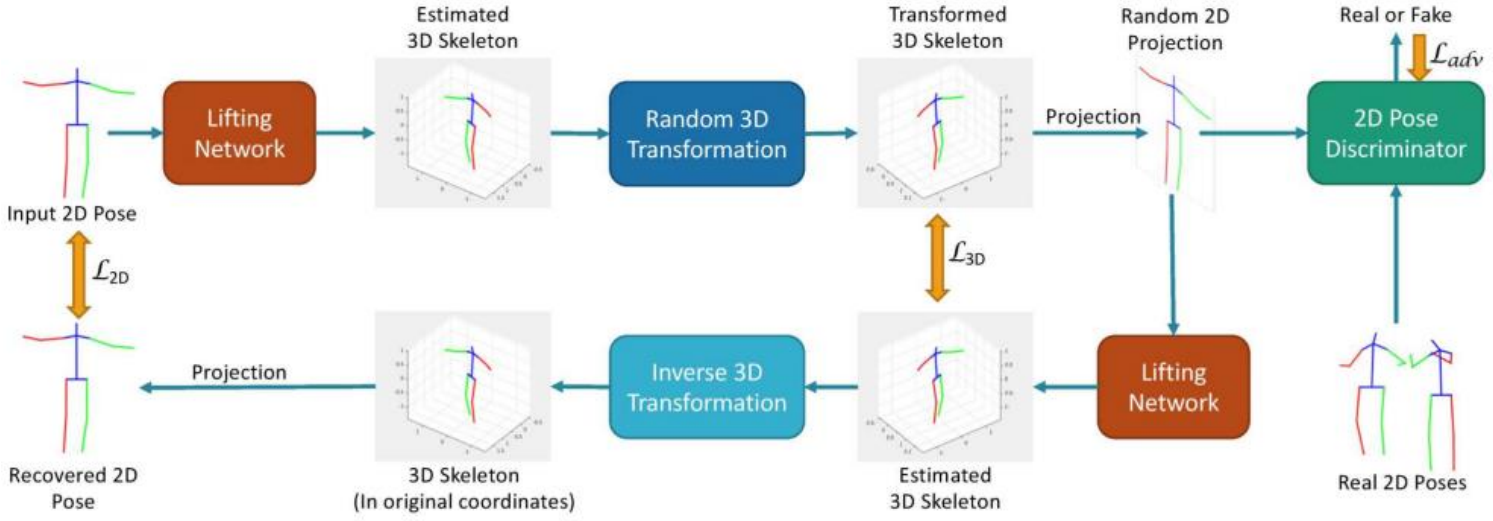
  - Unsupervised 3D Pose Estimation With Geometric Self-supervision – CVPR 2019

    - Contribution

      - 기하학적인 방법을 이용하여 리프팅을 이용해 discriminator의 loss를 이용해 성능을 향상 시킴
      - 2d domain adpatation 기술을 이용해 데이터에 적용해 성능을 향상 시킴
      - Temporal discriminator를 이용하여 training 하는 동안 2d-3d lifting의 성능을 향상 시킴

Development

- Based on CycleGAN
  - Model



Development

- New Idea

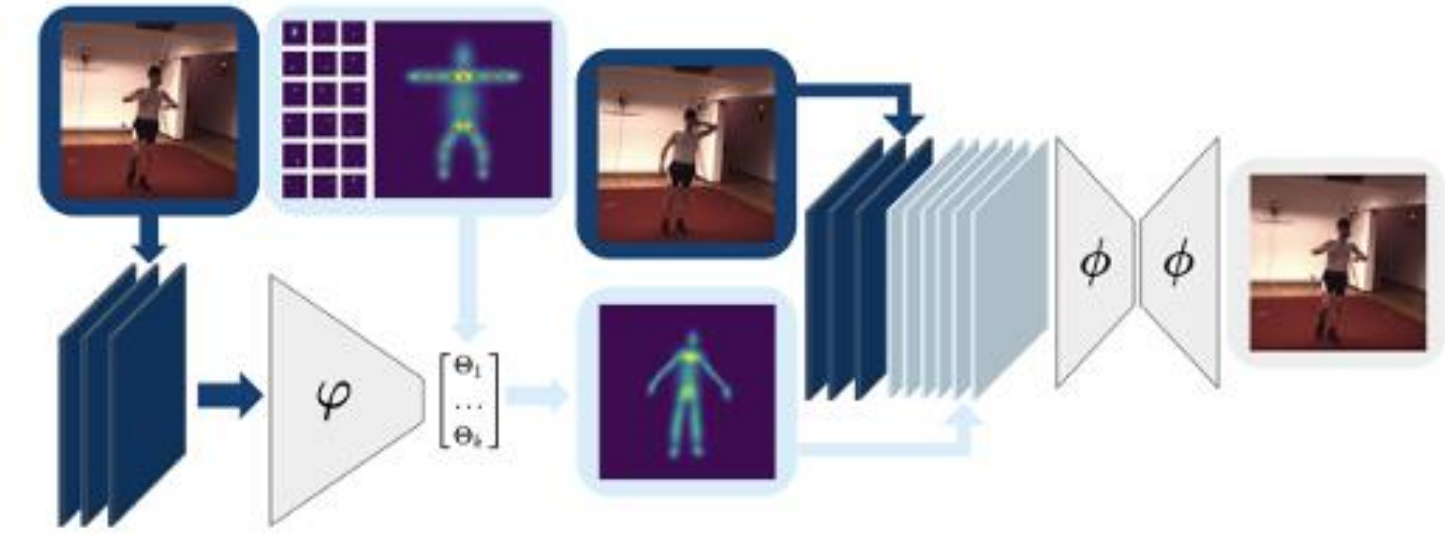
  - Unsupervised Human Pose Estimation Through Transforming shape templates – CVPR 2021

    - Contribution

      - 해석 가능한 2D human Keypoints를 기본으로 변경한 단일 2D 템플렛을 만듦
      - Labeled data를 하나도 사용하지 않고 unpaired한 데이터만 사용함
      - Wild한 유아 pose 추정 데이터 셋을 사용해서 평가함

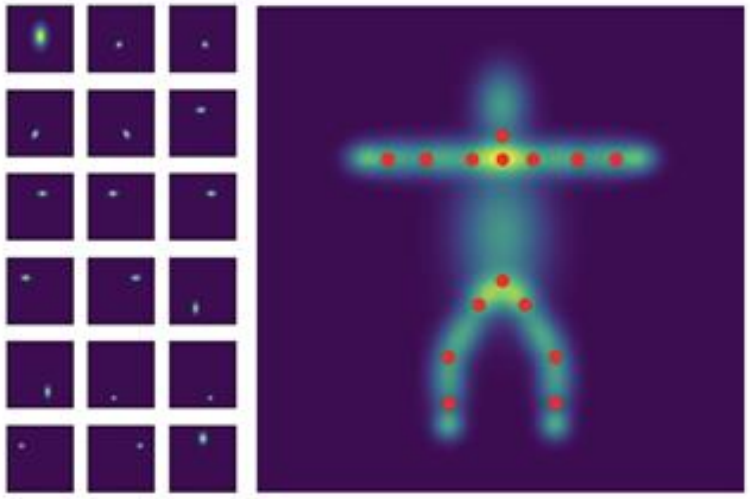
Development

- New Idea
  - Model



Development

- New Idea
  - Templates



– 2D Gaussian Distribution

Development

- New Idea
  - 3X3 Transformation Matrix

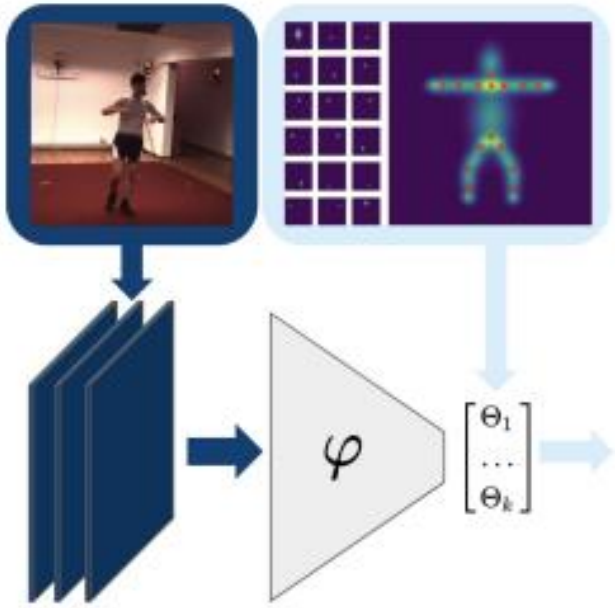
$$\Theta = \begin{bmatrix} \theta_{1,1} & \theta_{1,2} & t_x \\ \theta_{2,1} & \theta_{2,2} & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

Translations

Scale, rotation, shear

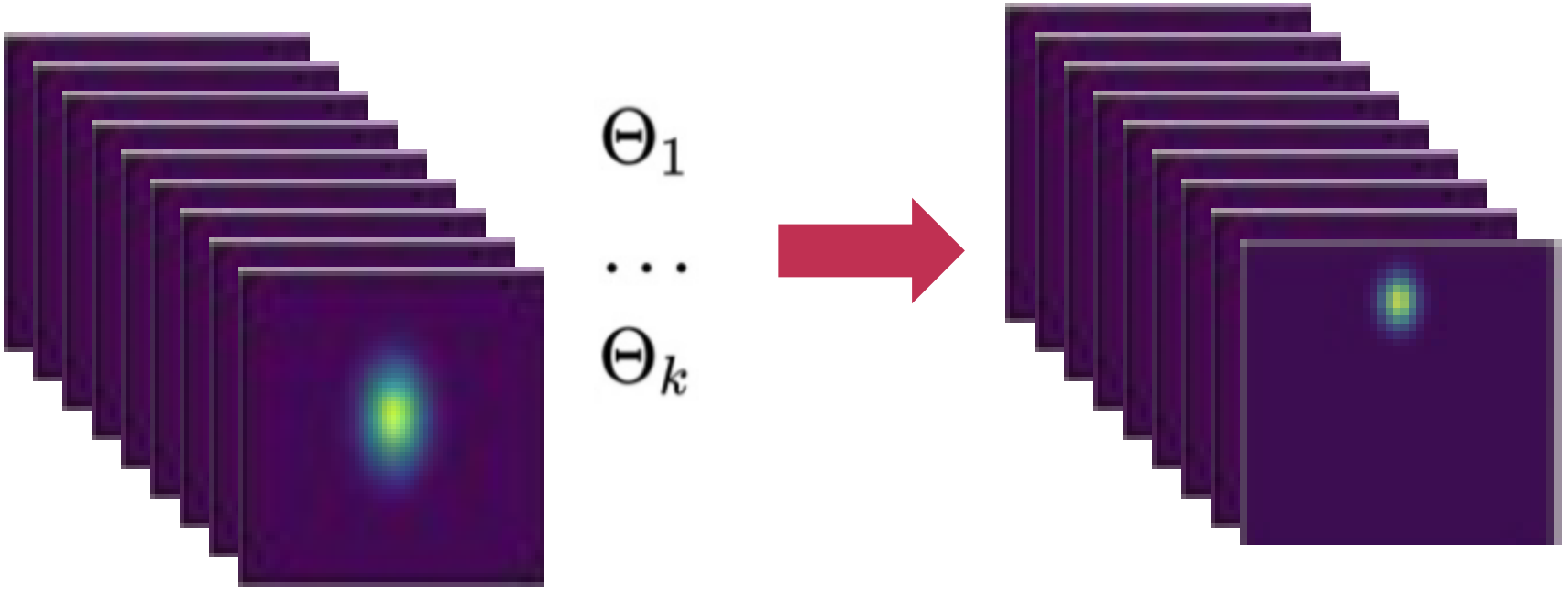
Development

- New Idea
  - Step 1



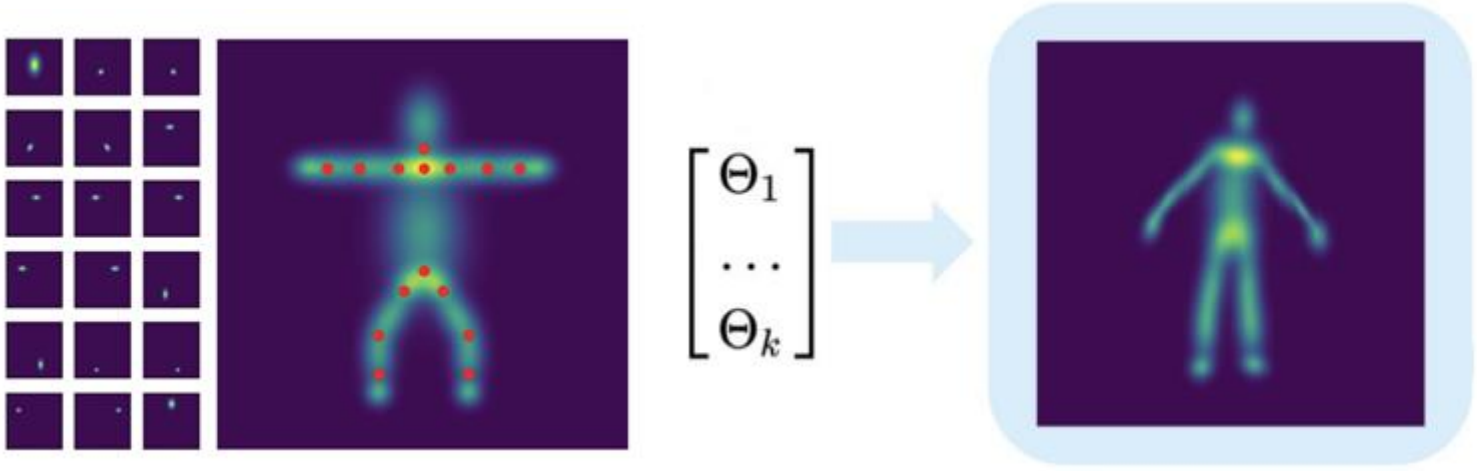
Development

- New Idea
  - Step 1



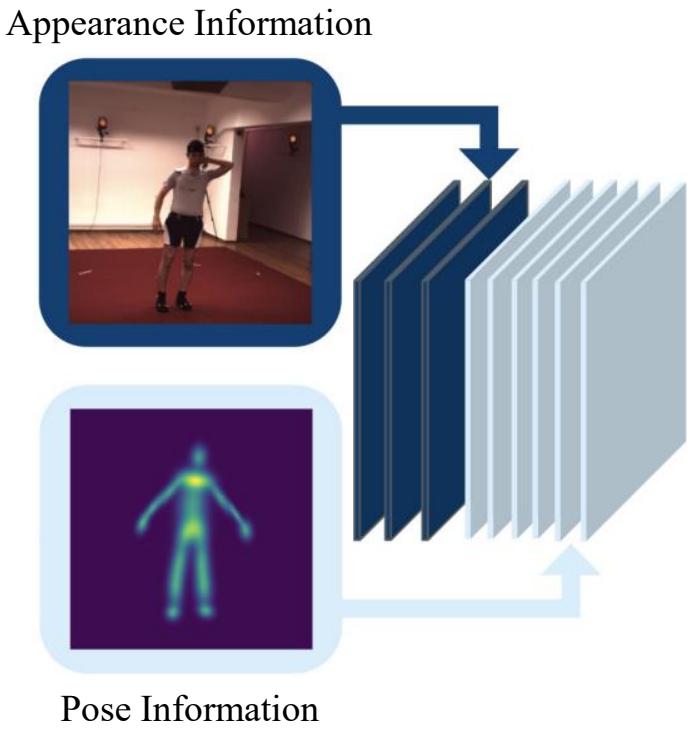
Development

- New Idea
  - Step 2
    - Core, left hip, right hip, left thigh, right thigh, left shin, right shin, left shoulder, right shoulder
    - Left arm, right arm, left forearm, right forearm, left foot, right foot, left hand, right hand, head



Development

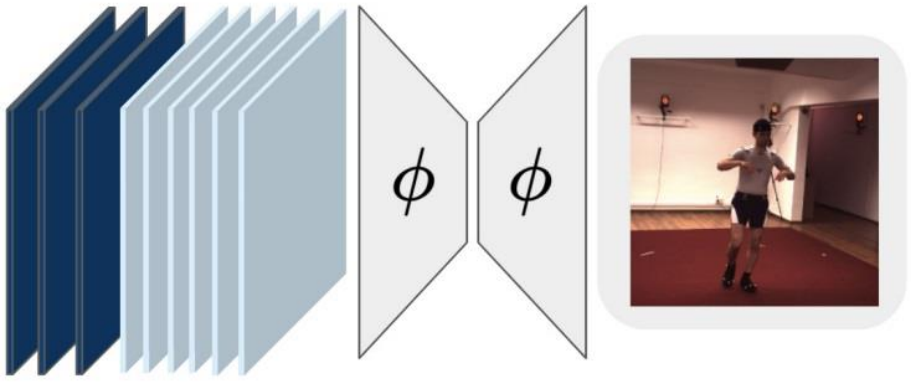
- New Idea
  - Step 3



Development

- New Idea
  - Step 4

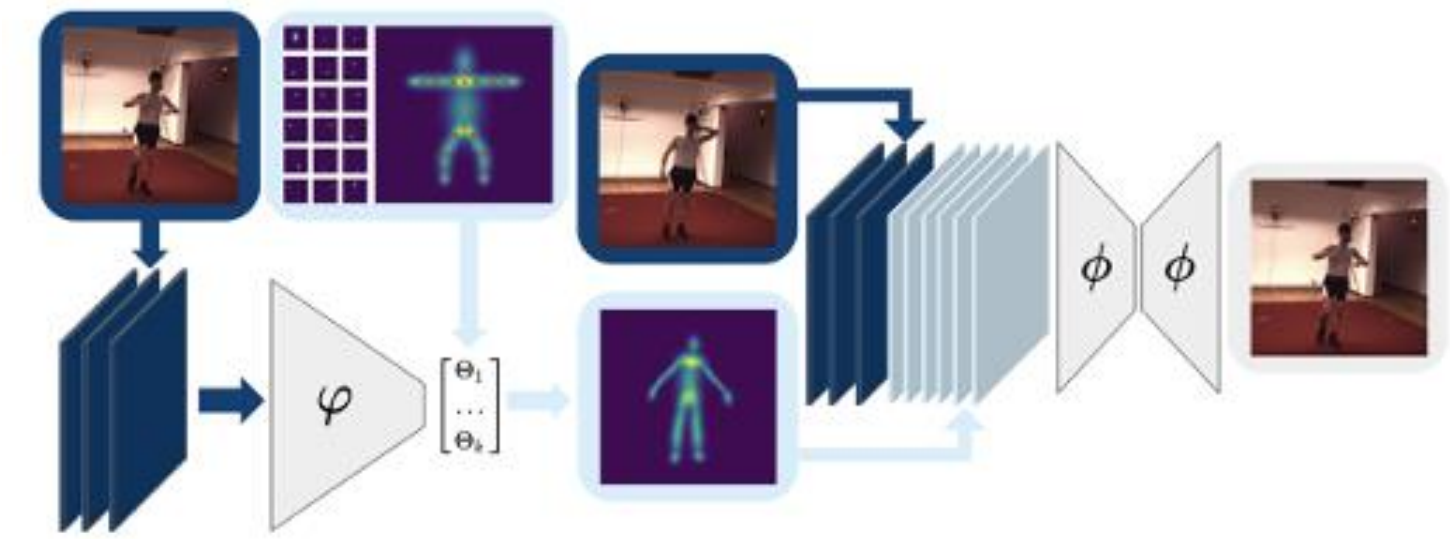
Appearance Information



Pose Information

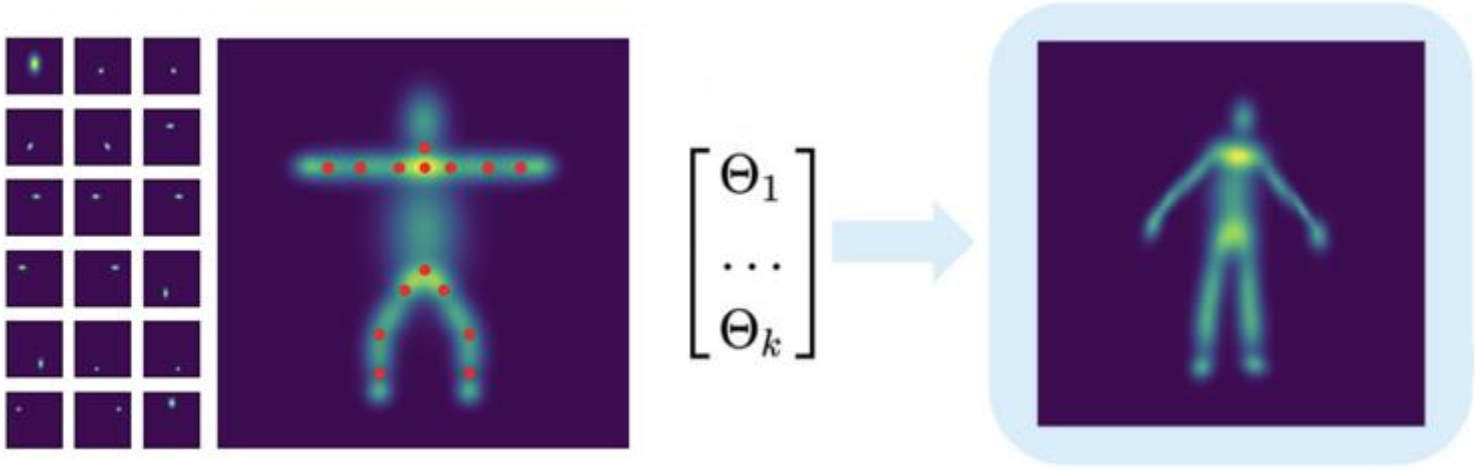
Development

- New Idea
  - Model



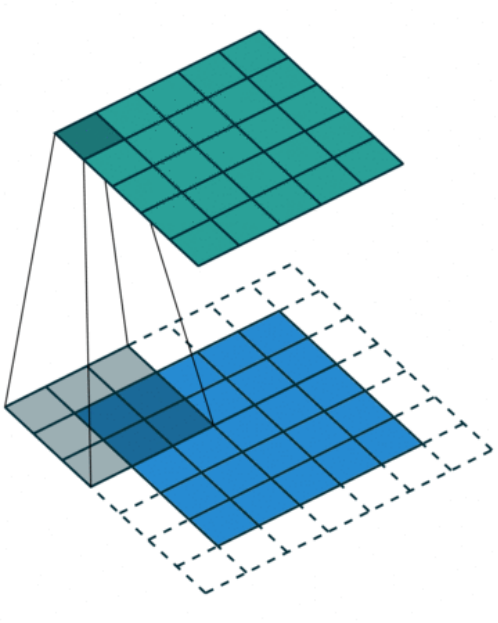
Development

- New Idea
  - Problem => 훈련이 가능한가?

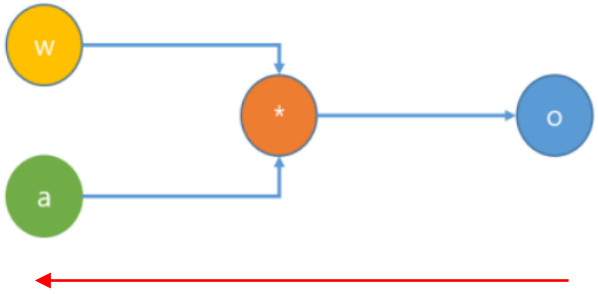


Development

- New Idea
  - Problem => 훈련이 가능한가?



Convolution



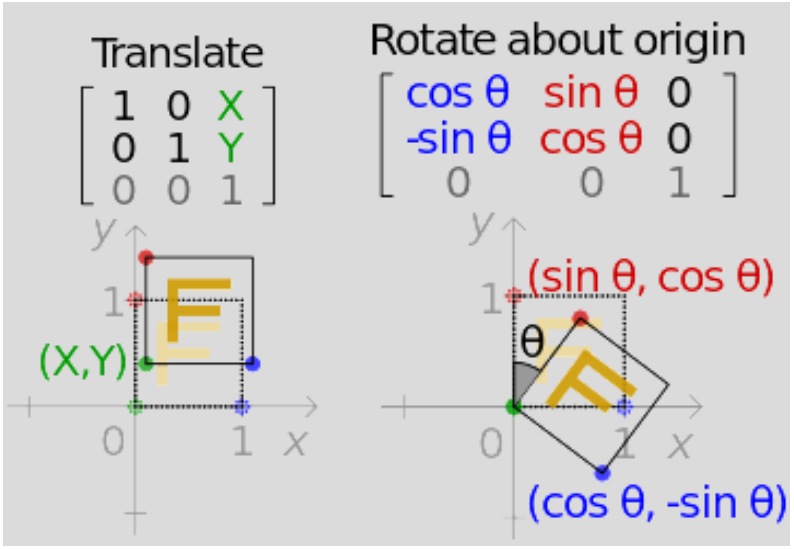
Backpropagation
Chain Rule

Development

- New Idea
  - Problem => 훈련이 가능한가?

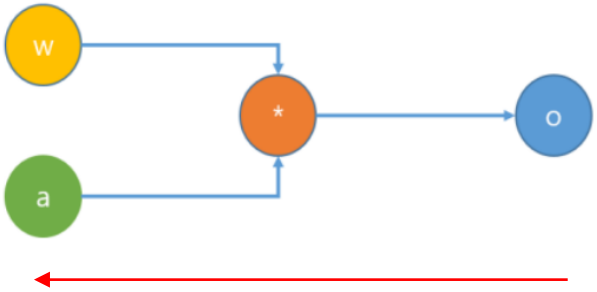
$$\begin{bmatrix} x` \\ y` \\ 1 \end{bmatrix} = \begin{bmatrix} \theta_{1,1} & \theta_{1,2} & t_x \\ \theta_{2,1} & \theta_{2,2} & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Destination Pixel



Development

- New Idea
  - Problem => 훈련이 가능한가?
    - Problem 1

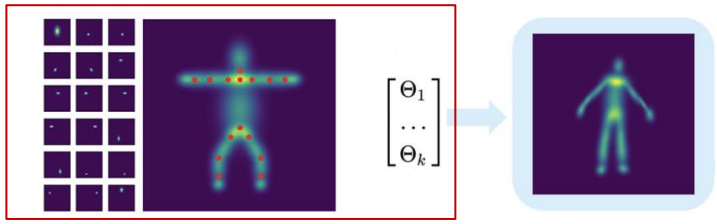


Backpropagation
Chain Rule

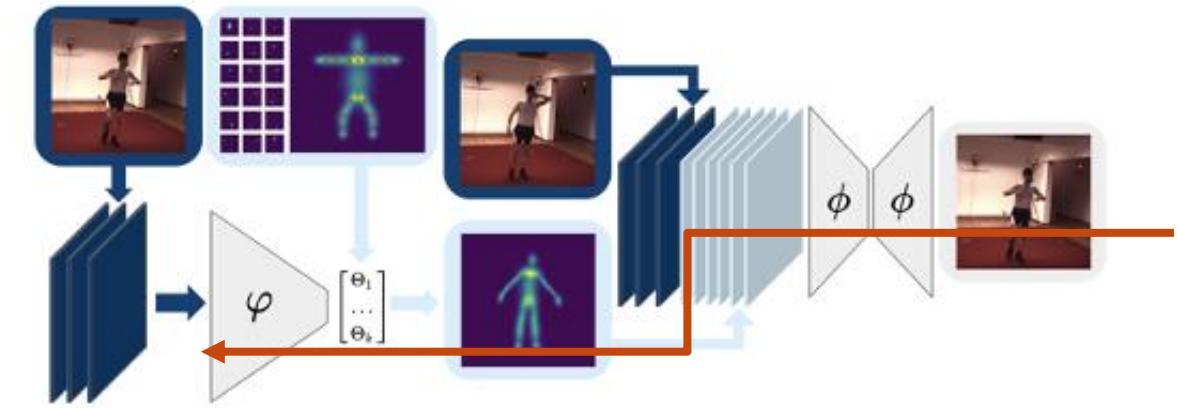
Transformation을 적용했을 때
Destination Pixel만 바꾸기 때문에
미분값 x

Development

- New Idea
  - Problem => 훈련이 가능한가?
    - Problem 2



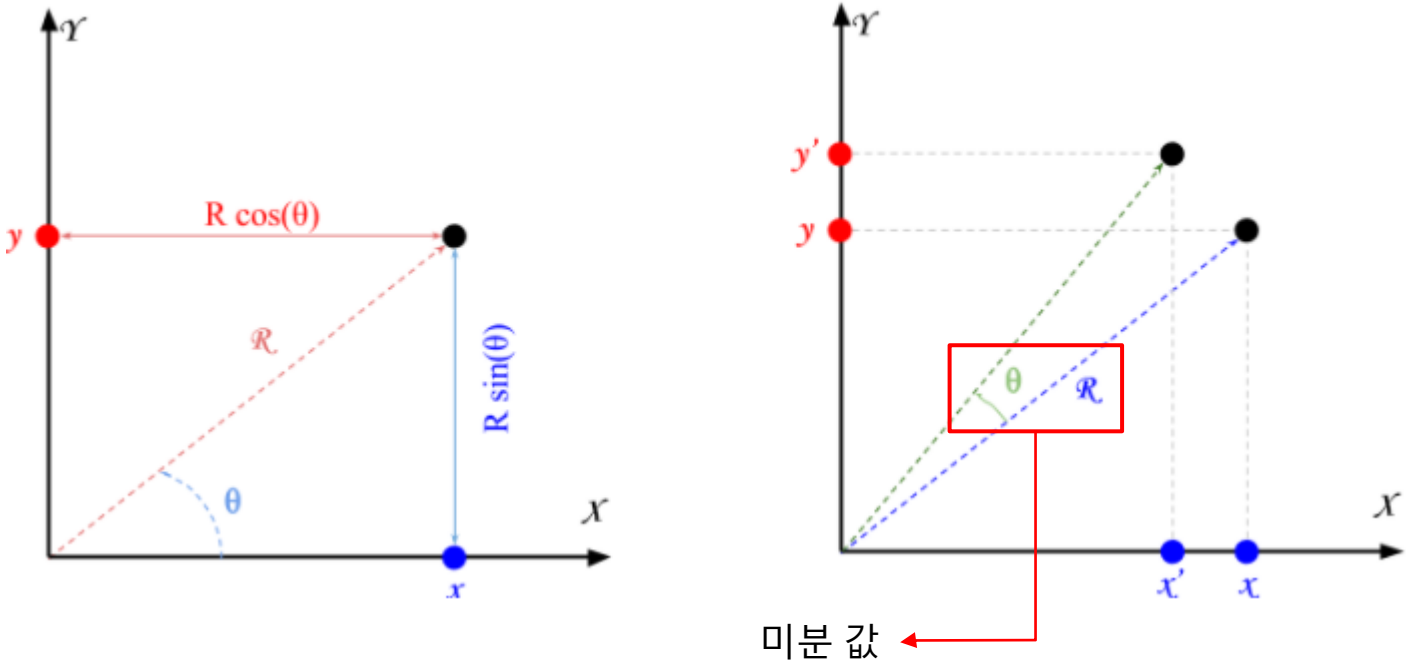
오른쪽 템플릿이 loss로 인해 미분 값으로 변할 때 Transformation matrix이 오른쪽 템플릿과 곱해진 형태여야 Transformation Matrix의 미분 값을 구할 수 있다.



backpropagation

Development

- New Idea
  - Problem => 훈련이 가능한가?
    - Kornia toolkit을 이용해서 훈련이 가능하도록 만듦



Development

- New Idea
  - Loss
    - Reconstruction Loss

$$\mathcal{L}_{recon} = ||\psi(\tilde{\mathbf{f}}_t) - \psi(\mathbf{f_t})||_1^1$$

VGG Pretrained Model Output Feature vector

    - Anchor-Point Loss

$$\mathcal{L}_{\text{anchor}} = \frac{1}{M} \sum_{\tilde{\mathbf{a}}_j^l, \tilde{\mathbf{a}}_k^m \in A} ||\tilde{\mathbf{a}}_j^l - \tilde{\mathbf{a}}_k^m||_2^2$$

Development

- New Idea
  - Anchor-Point Loss
    - Anchor Point
      - Ex) left Ankle – **left knee <Anchor Point>**
    - Anchor Point는 Transformation Matrix를 적용해도 위치
      - 최대 3개까지
  - Boundary Loss

$$\mathcal{L}_{bx} = \begin{cases} |a^j_{i,x}|, & \text{if } |a^j_{i,x}| > B \\ 0, & \text{otherwise,} \end{cases}$$

- Anchor Point는 이미지 안에 있어야 하므로 위의 식을 만족해야 한다.

Development

- New Idea
  - Combined Loss Formulation

$$\mathcal{L} = \mathcal{L}_{\text{recon}} + \lambda_1 \mathcal{L}_{\text{anchor}} + \lambda_2 (\mathcal{L}_{bx} + \mathcal{L}_{by})$$

Development

- New Idea
  - Result

Infants	all	hips	knees	feet	shoulders	hands	params
fully supervised (fine-tuned) baseline							
Xiao [60]	1.74	2.39	1.50	1.47	1.76	1.59	34.0 M
self-supervised (unpaired labels)							
Jakab [24]	8.98	6.89	8.18	13.15	5.33	11.36	8.6 M
self-supervised (template, no labels)							
Ours	4.86	3.79	4.60	5.53	3.19	7.21	7.8 M

H36M	all	wait	pose	greet	direct	discuss	walk
fully supervised baseline							
Newell [37]	2.16	1.88	1.92	2.15	1.62	1.88	2.21
self-supervised + supervised post-processing							
Thewlis [52]	7.51	7.54	8.56	7.26	6.47	7.93	5.40
Zhang [63]	4.14	5.01	4.61	4.76	4.45	4.91	4.61
Lorenz [34]	2.79	–	–	–	–	–	–
self-supervised (unpaired labels)							
Jakab [24]	2.73	2.66	2.27	2.73	2.35	2.35	4.00
self-supervised (template, no labels)							
Ours	3.31	3.51	3.28	3.50	3.03	2.97	3.55

- Infants - %-MSE normalized by image size is reported on a per body-part basis
- H36m - %-MSE normalized by image size is reported on a per action basis
- H36m에서 정확도가 더 낮은 이유는 jakab[24]에서는 약간의 unpaired한 label 데이터를 적용함

Development

- New Idea
  - Result

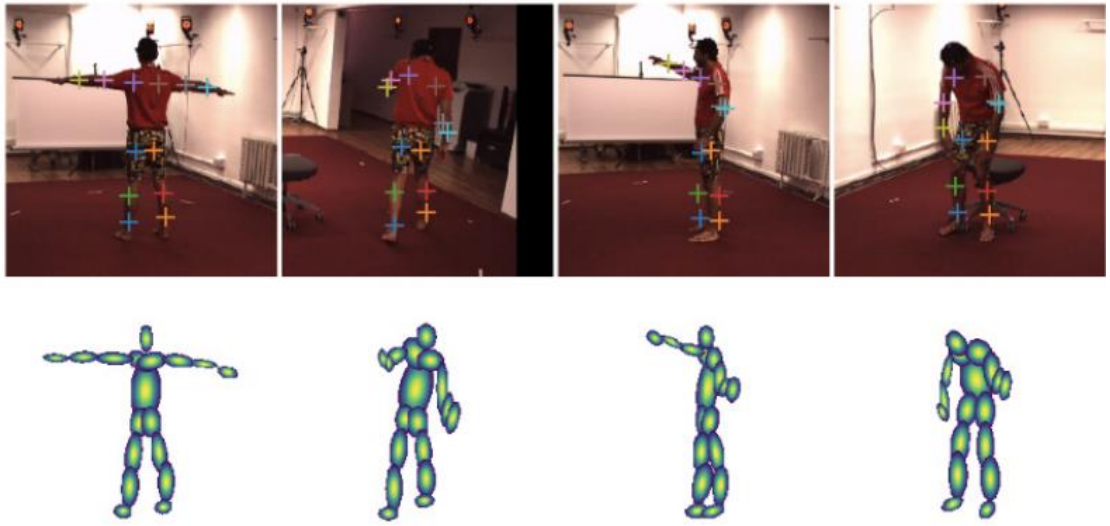
all	hips	knees	feet	shoulders	hands
anchor and boundary loss (ours)					
6.54	3.56	5.19	8.90	4.70	7.95
with anchor, no boundary loss					
65.99	64.65	71.70	76.33	59.00	62.59
with boundary, no anchor loss					
12.42	13.84	13.17	7.29	12.94	8.76
no anchor, no boundary loss (model diverges)					
446.47	433.64	385.77	662.41	502.69	276.49

Synthetic Infants	all	hips	knees	feet	shoulders	hands
self-supervised (unpaired labels)						
Jakab [24]	59.7	10.1	58.4	73.3	58.0	101.5
self-supervised (template, no labels)						
Ours	44.7	8.0	57.5	80.9	35.6	78.7

- Left – 각 Loss를 추가하거나 뺐을 때의 Mean joint distance in % of image size 값들
- Right 3D lifting result

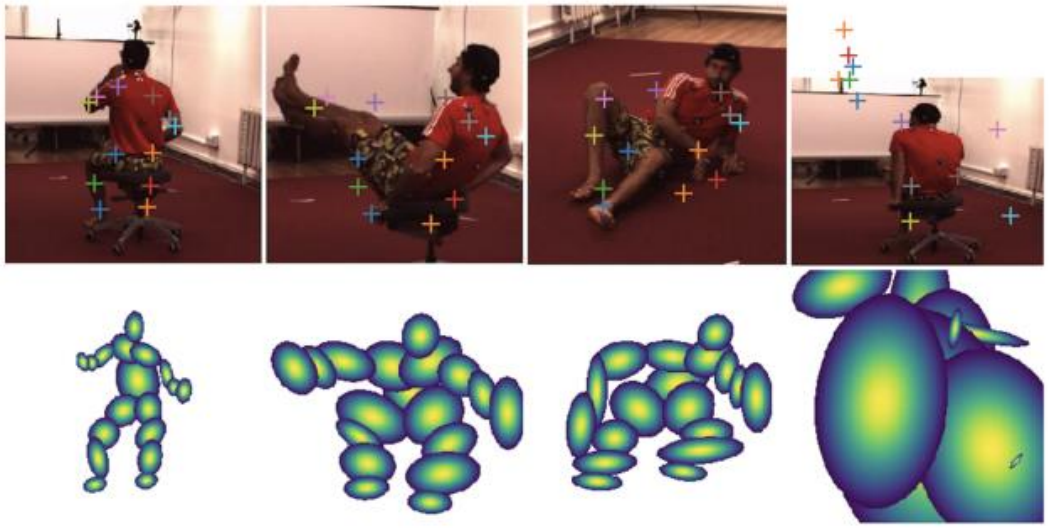
Development

- New Idea
  - Result



Development

- New Idea
  - Limitations



Reference

1. Luca Schmidtke, “Unsupervised Human Pose Estimation through Transforming Shape Templates”, CVRP 2021

2. Helge Rhodin, “Unsupervised Geometry-Aware Representation for 3D Human Pose Estimation”, ECCV 2018

3. Ching-Hang Chen. “Unsupervised 3D Pose Estimation with Geometric Self-Supervision”, CVRP 2019