

# 2D Articulated Human Pose Estimation

## Using Explainable Artificial Intelligence

André Oskar Andersen  
wpr684

Datalogisk Institut, Københavns Universitet

2021

# Introduktion

- ▶ *Articulated Human Pose Estimation og explainable artificial intelligence*



# Introduktion

- ▶ *Articulated Human Pose Estimation* og *explainable artificial intelligence*
- ▶ Få kilder udforsker pose estimation algoritmer

# Introduktion

- ▶ *Articulated Human Pose Estimation* og *explainable artificial intelligence*
  - ▶ Få kilder udforsker pose estimation algoritmer
- ▶ Hvorfor gør brug af XAI?
    1. Forbedrer præstation
    2. Bygger tillid - styrker og svagheder
    3. Vi kan lære af modellen

# Introduktion

- ▶ *Articulated Human Pose Estimation* og *explainable artificial intelligence*
- ▶ Få kilder udforsker pose estimation algoritmer

- ▶ Hvorfor gør brug af XAI?

1. Forbedrer præstation
2. Bygger tillid - styrker og svagheder
3. Vi kan lære af modellen

- ▶ Problem definition

1. Implementer Stacked Hourglass af Newell *et al.*
2. Udforsk Stacked Hourglass
3. Modifier Stacked Hourglass vha. viden

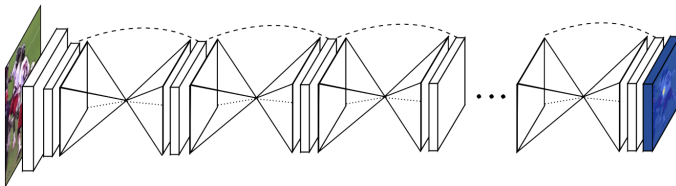
# Data

- ▶ 2017 Microsoft COCO
- ▶ Træning + validering:  
69.000 billeder
- ▶ Flere personer i hvert billede
- ▶ 17 keypoints per person
- ▶ Centrerer billede omkring  
hver person
- ▶ Laver 17 heatmaps



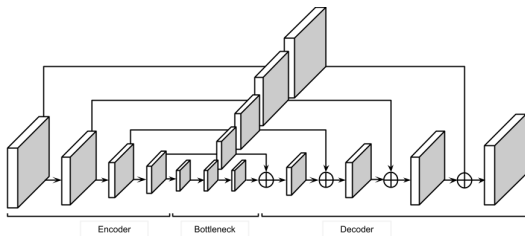
# Stacked hourglass

- ▶ Stacked hourglass
  - ▶ Stakker hourglasses
  - ▶ Input:  $256 \times 256$  billede der downsample inden første hourglass
  - ▶ Output:  $17 \times 64 \times 64$  heatmaps



# Stacked hourglass

- ▶ Stacked hourglass
  - ▶ Stakker hourglasses
  - ▶ Input:  $256 \times 256$  billede der downsample inden første hourglass
  - ▶ Output:  $17 \times 64 \times 64$  heatmaps
- ▶ Hourglass
  - ▶ Residuale modules





# Stacked hourglass

- ▶ Stacked hourglass
  - ▶ Stakker hourglasses
  - ▶ Input:  $256 \times 256$  billede der downsample inden første hourglass
  - ▶ Output:  $17 \times 64 \times 64$  heatmaps
- ▶ Hourglass
  - ▶ Residuale modules
- ▶ Loss- og activationfunction

# Eksperiment og resultat

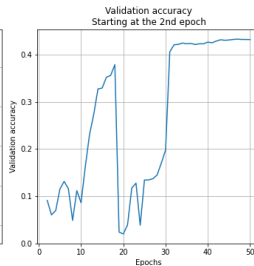
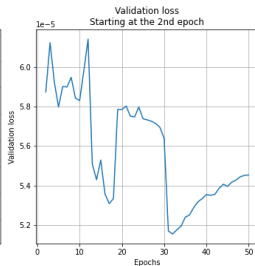
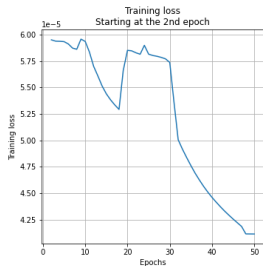
- ▶ Kun ét hourglass

# Eksperiment og resultat

- ▶ Kun ét hourglass
- ▶ Følger ellers Newell *et al.* og Camilla Olsen

# Eksperiment og resultat

- ▶ Kun ét hourglass
- ▶ Følger ellers Newell *et al.* og Camilla Olsen
- ▶ Overfit

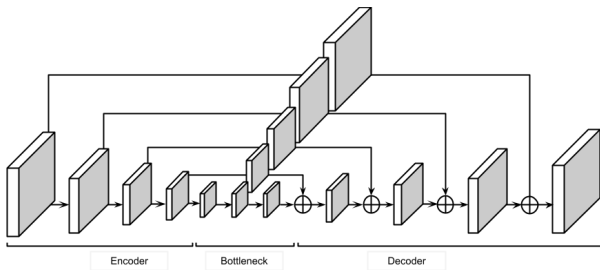


# Eksperiment og resultat

- ▶ Kun ét hourglass
- ▶ Følger ellers Newell *et al.* og Camilla Olsen
- ▶ Overfit
- ▶ Validation PCK accuracy: 0.433. Test PCK accuracy: 0.441

# Fortolkning af modellen 1 - Effekt af skip-connections

- Påstand: anvendes til at "redde" information

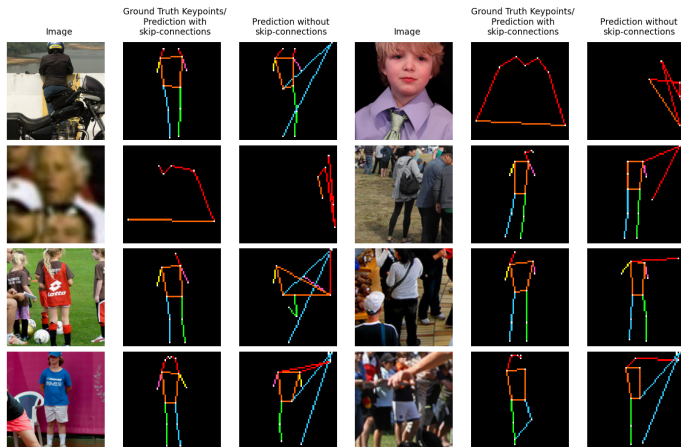


# Fortolkning af modellen 1 - Effekt af skip-connections

- ▶ Påstand: anvendes til at "redde" information
- ▶ Fremgang: SHG med skip-connection vs SHG uden skip-connection

# Fortolkning af modellen 1 - Effekt af skip-connections

- ▶ Påstand: anvendes til at "redde" information
- ▶ Fremgang: SHG med skip-connection vs SHG uden skip-connection
- ▶ Resultat





## Fortolkning af modellen 2 - Effekt af principal komponenter

- ▶ Formål: effekt af *latent spaces* principal komponenter

# Fortolkning af modellen 2 - Effekt af principal komponenter

- ▶ Formål: effekt af *latent spaces* principal komponenter
- ▶ PCA og "Gå" ud af gennemsnitspunkt med stepsize =  $c \cdot \sqrt{\lambda}$

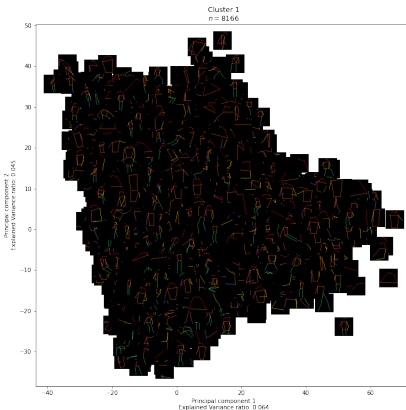
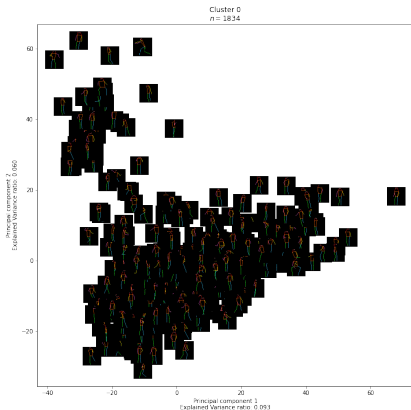


## Fortolkning af modellen 3 - Clustering af latent space

- ▶ Formål: Har modellen lært at relatere lignende datapunkter til hinanden?

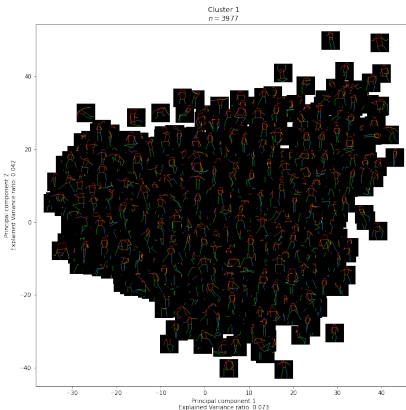
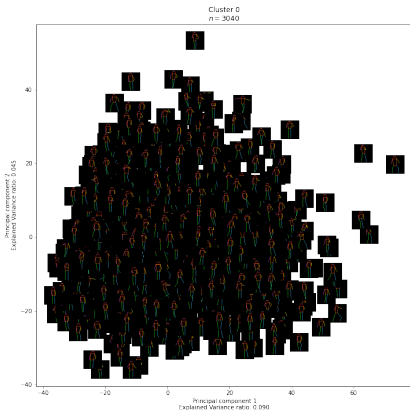
# Fortolkning af modellen 3 - Clustering af latent space

- ▶ Formål: Har modellen lært at relatere lignende datapunkter til hinanden?
- ▶ Resultater



# Fortolkning af modellen 3 - Clustering af latent space

- ▶ Formål: Har modellen lært at relatere lignende datapunkter til hinanden?
- ▶ Resultater



# Modificering af Stacked hourglass

- ▶ Formål: Anvende viden fra XAI til at optimere model

# Modificering af Stacked hourglass

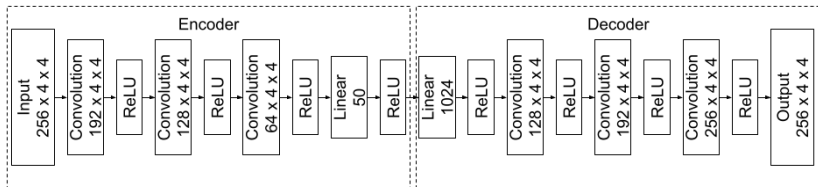
- ▶ Formål: Anvende viden fra XAI til at optimere model
- ▶ Forbedringspunkter:
  1. Støj
  2. Misklassificeringer



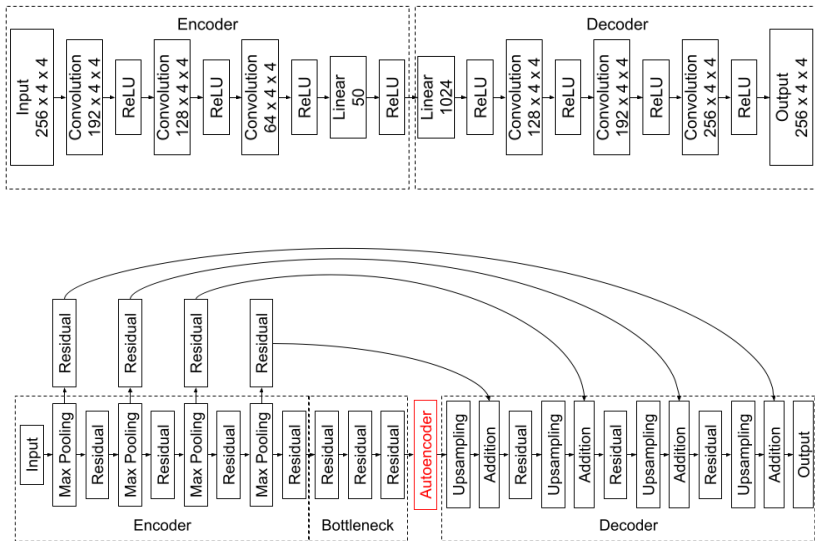
# Modificering af Stacked hourglass

- ▶ Formål: Anvende viden fra XAI til at optimere model
- ▶ Forbedringspunkter:
  1. Støj
  2. Misklassificeringer
- ▶ Løsning: Autoencoder

# Modificering af Stacked hourglass

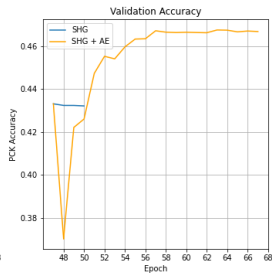
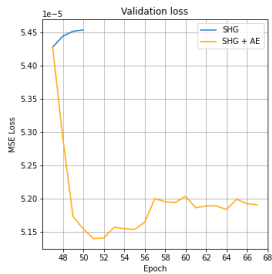
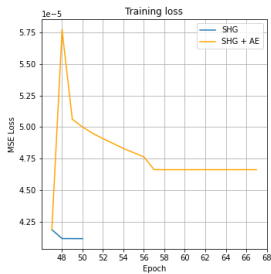


# Modificing af Stacked hourglass



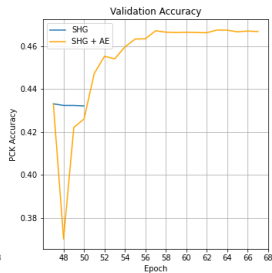
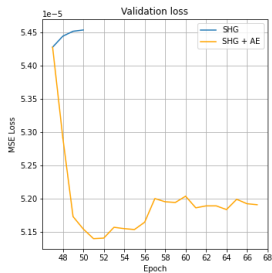
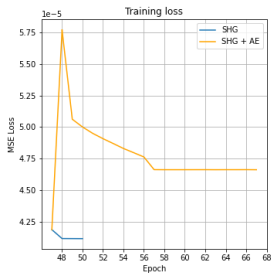
# Resultat

## ► Bedre performance



# Resultat

- Bedre performance
- Plateau



# Resultat

- ▶ Bedre performance
- ▶ Plateau
- ▶ Validation PCK accuracy: 0.467 (+7.8%/0.034)
- ▶ Testing PCK accuracy: 0.474 (+7.48%/0.033)

# Diskussion 1 - Sammenligning af modeller

► Camilla Olsen (kun synlige keypoints)

Beskrivelse	# Stakke	Testing accuracy
Olsen - 2A	2	0.72
Olsen - 2B	2	0.81
Olsen - 2M	2	0.83
Min SHG	1	0.469
<b>Min modificeret SHG</b>	<b>1</b>	<b>0.576</b>

# Diskussion 1 - Sammenligning af modeller

► Camilla Olsen (kun synlige keypoints)

Beskrivelse	# Stakke	Testing accuracy
Olsen - $2A$	2	0.72
Olsen - $2B$	2	0.81
Olsen - $2M$	2	0.83
Min SHG	1	0.469
<b>Min modificeret SHG</b>	<b>1</b>	<b>0.576</b>

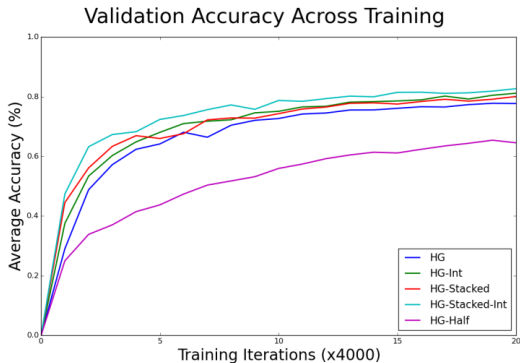
► Forskelle

1. Antal stakke
2. Trænet kun på synlige keypoints



# Diskussion 1 - Sammenligning af modeller

- ▶ Newell *et al.*
  - ▶ Ét hourglass: keypoints ikke forbundet til overkrop/hoved under træning: 0.65
  - ▶ Mine:
    1. Standard SHG: 0.32
    2. Modificeret SHG: 0.38



# Diskussion 1 - Sammenligning af modeller

- ▶ Newell *et al.*
  - ▶ Ingen accuracy for alle keypoints, dog for keypoints ikke forbundet til overkrop/hoved, maximeres løbende accuracy ved 0.65
  - ▶ Min:
    1. Standard SHG: 0.32
    2. Modificeret SHG: 0.38
- ▶ Forskelle:
  1. Forskellig data
  2. Forskelligt setup skyldet manglende information

# Diskussion 2 - Hvorfor hjalp autoencoderen?

## 1. Clustering

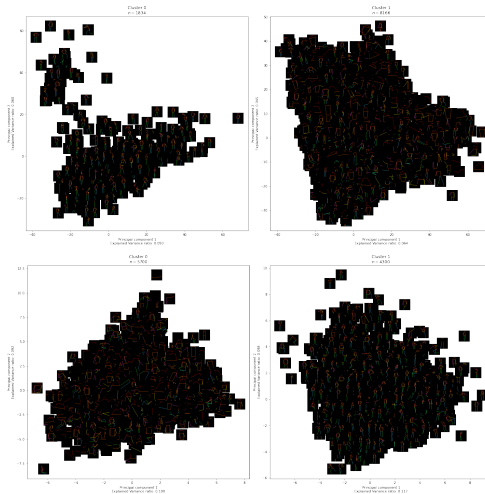


Figure: Top: SHG. Bund: SHG + AE

# Diskussion 2 - Hvorfor hjælp autoencoderen?

1. Clustering
2. Fjernet støj i latent space

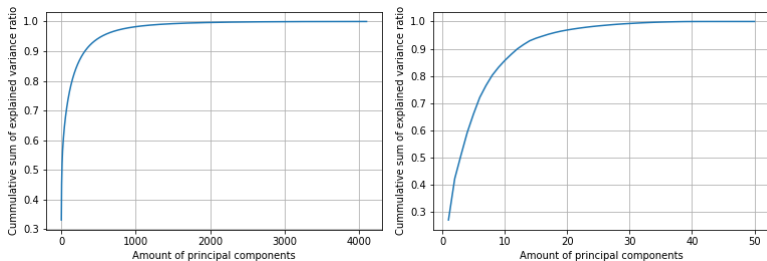


Figure: Venstre: SHG. Højre: SHG + AE

## Future work

1. Effekt af at stakke flere modificeret hourglasses

## Future work

1. Effekt af at stakke flere modificeret hourglasses
2. Redudans andre steder + introduktion af redudans ved flere hourglasses

## Future work

1. Effekt af at stakke flere modificeret hourglasses
2. Redudans andre steder + introduktion af redudans ved flere hourglasses
3. Forbedring ved yderligere korrektion af misklassification

# Konklusion

1. Implementeret Stacked hourglass



# Konklusion

1. Implementeret Stacked hourglass
2. Udforsket model
  - 2.1 Bekræftet påstand om skip-connections
  - 2.2 Effekt af første principal komponent, samt støj
  - 2.3 Separation

# Konklusion

1. Implementeret Stacked hourglass
2. Udforsket model
  - 2.1 Bekræftet påstand om skip-connections
  - 2.2 Effekt af første principal komponent, samt støj
  - 2.3 Seperation
3. Brugt opnået viden til at forbedre Stacked hourglass

# Fejl og rettelser

## 1. Normalfordeling

$$\mathcal{N}(\mu, \sigma^2)$$

# Fejl og rettelser

1. Normalfordeling
2. Nearest Neighbour Upsampling

---

**Algorithm 1** Nearest Neighbour Upsampling

---

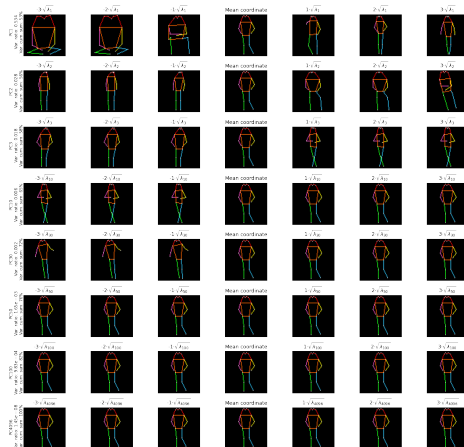
**Require:** Input image  $X$  of size  $m \times n$

**Require:** Wanted output size  $xm \times yn$ , where  $x, y \in \mathbb{Z}^+$

- 1: Create empty image  $O$  of size  $xm \times yn$
  - 2: **for all** pixel  $p \in X$  **do**
  - 3:      $i, j = \text{index of } p \text{ in } X$
  - 4:     Insert  $p$  at index  $(\lfloor xi \rfloor, \lfloor yj \rfloor)$  in  $O$
  - 5: **for all** empty pixel  $p \in O$  **do**
  - 6:     Let  $p$  be the value of the nearest neighbour
  - 7: **return**  $O$
-

# Fejl og rettelser

1. Normalfordeling
2. Nearest Neighbour Upsampling
3. Effekt af principal komponenter



# Fejl og rettelser

1. Normalfordeling
2. Nearest Neighbour Upsampling
3. Effekt af principal komponenter
4. Batch normalization bruges ikke for at undgå overfitting