

# Automated locomotion score from video

Deep learning keypoint detection in video images

2019-07-04 , Ard Nieuwenhuizen, Janne Kool

Breed4Food symposium

Wageningen UR, Agro Food Robotics



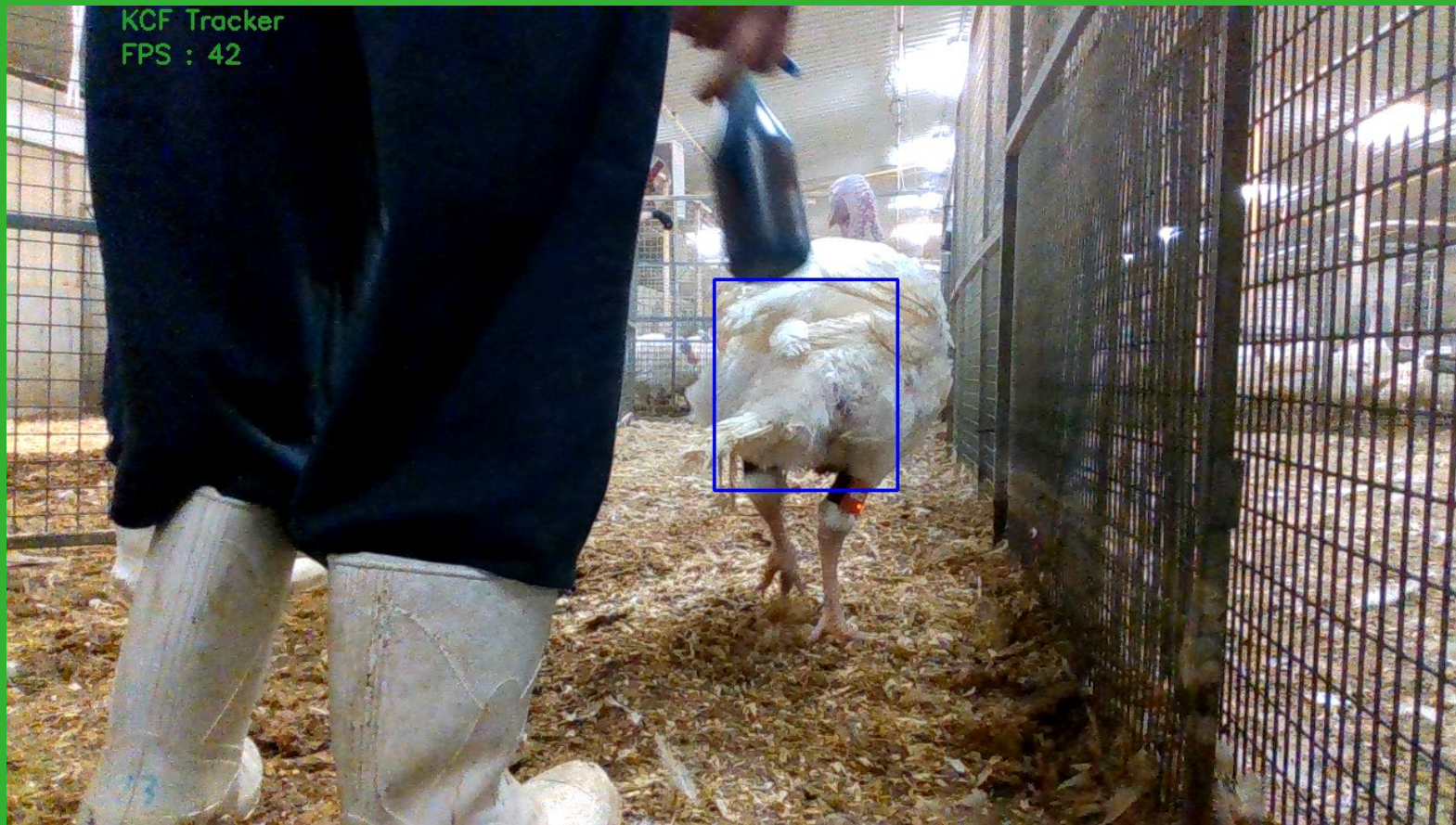
# Determine locomotion score from video

- Method development is the first step in data analysis
  - Separate the interesting objects
  - Determine properties or features of the objects
  - Gather the data
  - Compare with other scores
  - Summarize results
- Separation of objects and determination of features combined in **deep learning pose estimation**

# Classical approach - results for blob detection



# First results in analysis - tracking



# Classical vs deep learning pose estimation

- No static thresholds anymore
- User input on setting thresholds changes into annotation that can be generalised
- <https://alexemg.github.io/DeepLabCut/>
- Good tutorials, easy setup, well maintained online
- Training and classification on GPU, speeds up calculation times

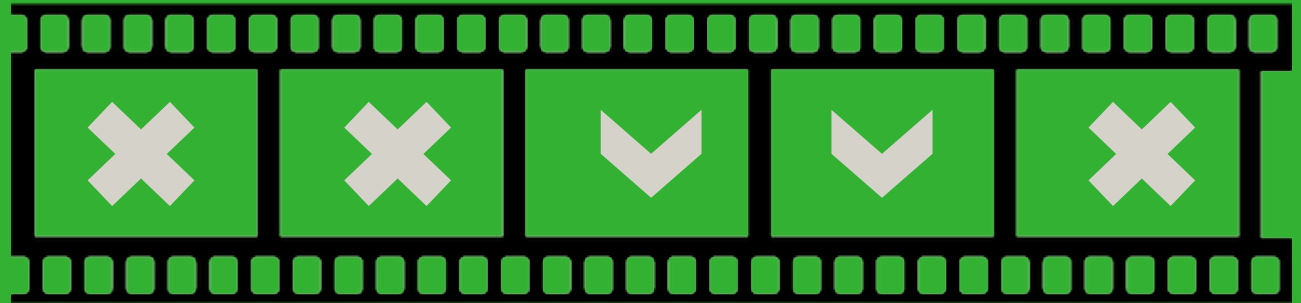
# How to implement pose estimation in video?

- Determine skeleton joint positions or bodyparts
- Feet L, R
- Heel L, R
- Knee L, R
- Neck
- Head



# Frame selection from movies for annotation

- From movies frames are selected by automatic algorithm, such that maximum variability is achieved in training and annotation data



# Annotation of data by Python GUI interaction - 1

The screenshot displays the DeepLabCut2.0 - Labeling ToolBox interface. The main window shows a video frame titled "6/19 img236.png" of a turkey in a barn. A vertical color bar legend on the right side of the image identifies body parts: Left food (blue), Left heel (light blue), Left knee (cyan), Right food (green), Right heel (yellow-green), Right knee (yellow), Neck (orange), and Head (red). The y-axis on the left of the image ranges from 0 to 700, and the x-axis at the bottom ranges from 0 to 500.

On the right side of the GUI, there is a control panel with the following options:

- Adjust marker size.
- Select a bodypart to label
  - Left food
  - Left heel
  - Left knee
  - Right food
  - Right heel
  - Right knee
  - Neck
  - Head

At the bottom of the GUI, there is a toolbar with buttons for "Load frames", "<<Previous", "Next>>" (highlighted), "Help", "Zoom", "Home", "Pan", "Lock View" (checkbox), "Save", and "Quit". A status bar at the very bottom indicates "Working on folder: 20180605\_105148".



# Annotation of data by Python GUI interaction - 2

DeepLabCut2.0 - Labeling ToolBox

7/19 img240.png

0 100 200 300 400 500 600 700

Left food  
Left heel  
Left knee  
Right food  
Right heel  
Right knee  
Neck  
Head

Adjust marker size.

Select a bodypart to label

- Left food
- Left heel
- Left knee
- Right food
- Right heel
- Right knee
- Neck
- Head

Load frames <<Previous Next>> Help Zoom Home Pan Lock View Save Quit

Working on folder: 20180605 105148

# Annotation data

- From each movie 20 frames were annotated, 12 movies total
- 8 clicks for each joint in a frame.  $8 \times 20 \times 12 = 1920$  clicks total
- This is very minimal with respect to the results achieved!

Annotation data looks like (pixel coordinates):

bodyparts	Left food	Left food	Left heel	Left heel	Left knee	Left knee	Right food	Right food	Right heel	Right heel	Right knee	Right knee	Neck	Neck	Head	Head
coords	x	y	x	y	x	y	x	y	x	y	x	y	x	y	x	y
labeled-data\20180605_121322\img013.png																
labeled-data\20180605_121322\img019.png																
labeled-data\20180605_121322\img020.png																
labeled-data\20180605_121322\img048.png																
labeled-data\20180605_121322\img049.png																
labeled-data\20180605_121322\img064.png																
labeled-data\20180605_121322\img105.png													308.296	153.3929	288.1398	82.84608
labeled-d	380.1026	638.4026	362.4659	567.8557	380.1026	503.6077	271.7628	671.1565	274.2824	582.9729	252.8663	517.4651	322.1534	274.3304	313.3351	215.1214
labeled-d	381.3624	639.6624	375.0636	569.1155	396.4796	512.4261	317.1144	628.3245	329.712	566.596	308.296	512.4261	356.1671	261.7328	352.3878	216.3812
labeled-d	373.8038	640.9221	359.9464	557.7776	383.8819	494.7894	323.4132	628.3245	323.4132	562.8167	309.5558	506.1272	362.4659	264.2523	356.1671	225.1996
labeled-d	468.2862	574.1546	455.6886	521.2444	470.8057	479.6722	396.4796	582.9729	398.9991	523.764	388.921	479.6722	421.6749	279.3695		
labeled-d	470.8057	576.6741	451.9093	522.5042	467.0265	480.9319	393.9601	581.7132	397.7394	517.4651	395.2198	480.9319	427.9737	269.2914		
labeled-d	434.2726	502.3479	433.0128	460.7757	440.5714	434.3206	390.1808	469.594	402.7784	453.2171	386.4015	434.3206	406.5577	308.3441	405.2979	283.1488
labeled-d	402.7784	472.1136	401.5187	453.2171	400.2589	433.0608	375.0636	477.1526	375.0636	454.4769	364.9855	433.0608	383.8819	329.7601		
labeled-data\20180605_121322\img349.png																
labeled-data\20180605_121322\img364.png																

# Training of the algorithm after annotation

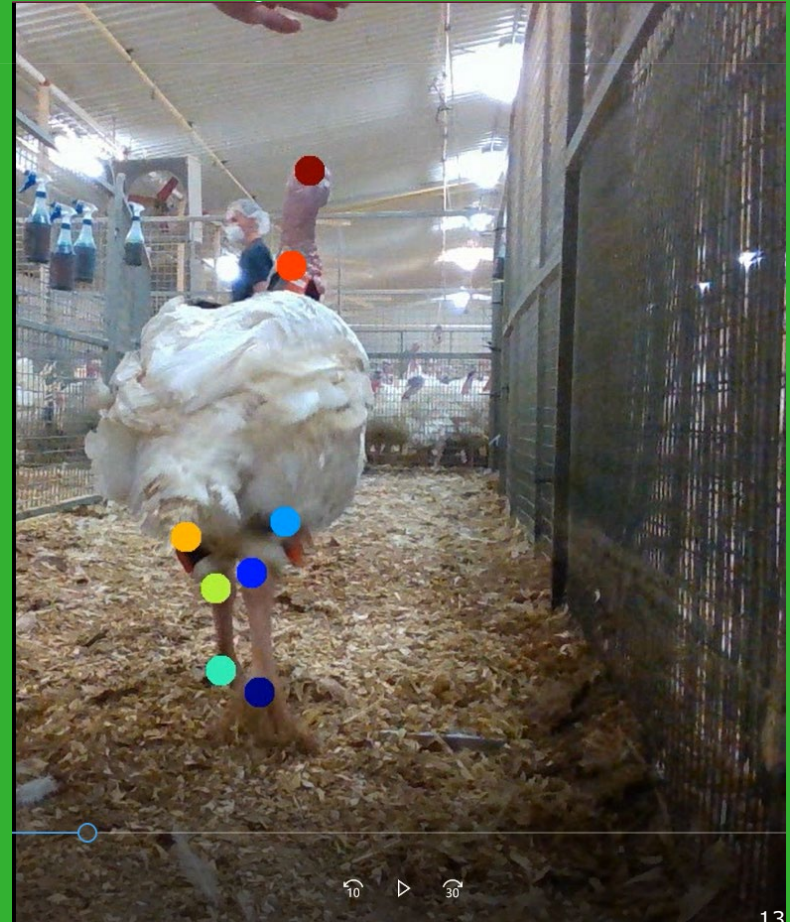
- Deep learning GPU machine
  - Trained overnight and during weekend
- Can also be done on HPC Annuna from WUR



- Do not look at the size of the markers, they do not scale
- It is all about the position of the joints that have been detected



# Result movies



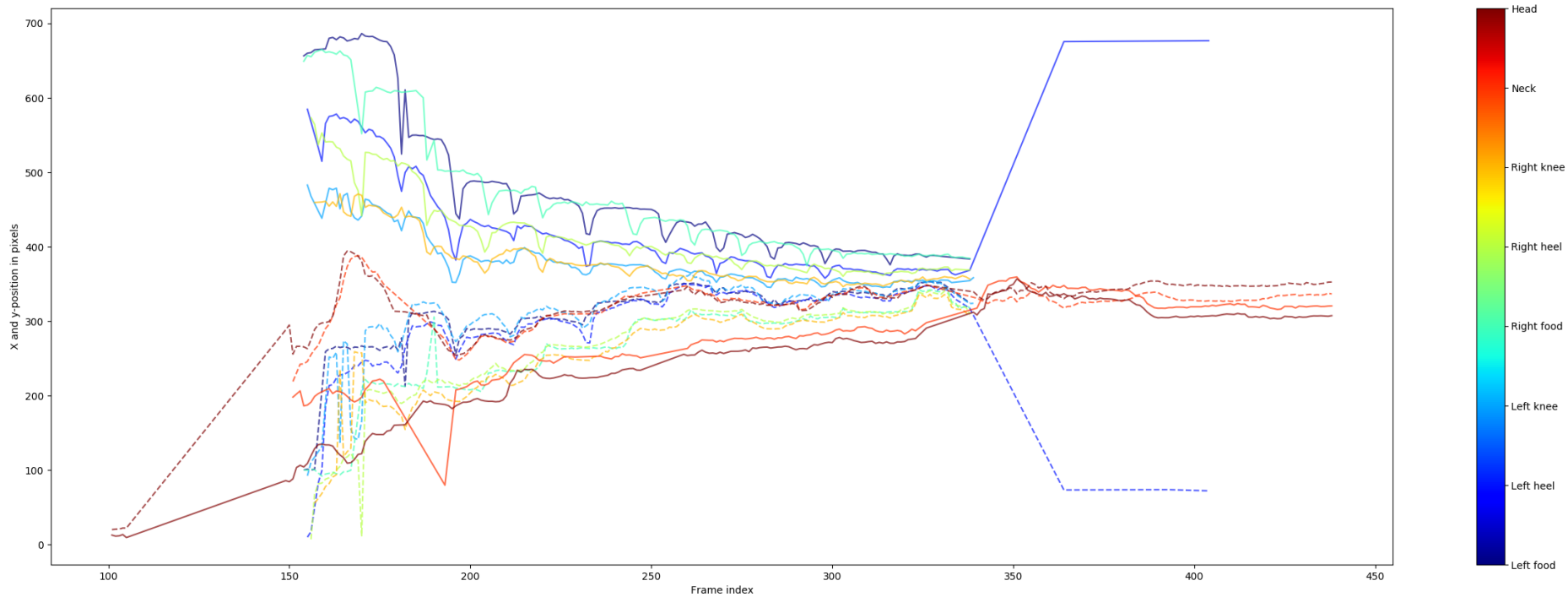
# Moving objects do not always interfere!



- This is very good!
- Still complete skeleton available

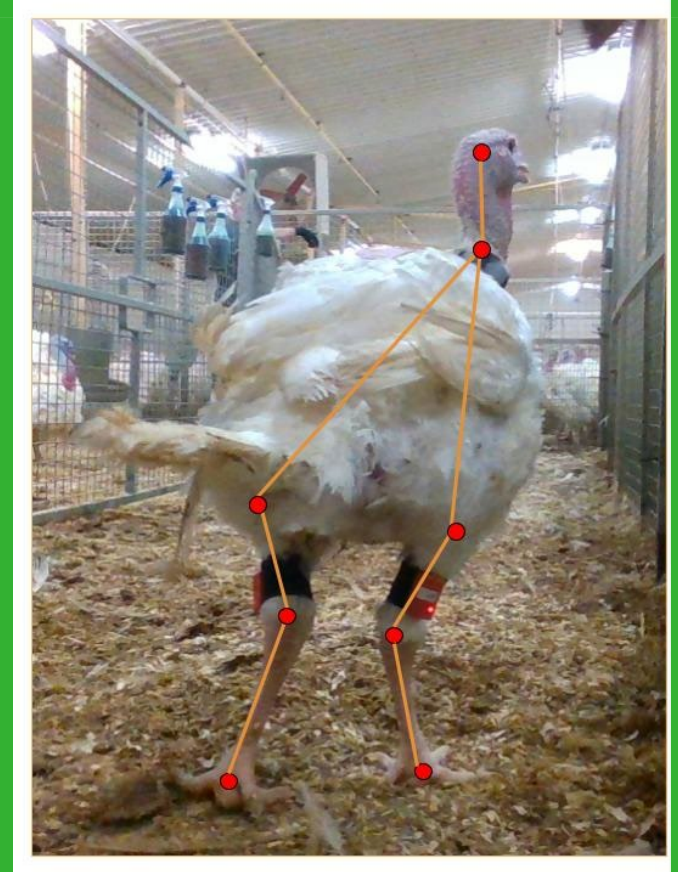
# Time series of position of joint through movie

- Lines converge, due to scaling



# Next steps in analysis

- Connect the joints when skeleton is complete
- Determine scale invariant parameters, e.g. angles, angle changes
- Prepare time series, identical to step analysis





# Questions and suggestions

Ard Nieuwenhuizen

[ard.Nieuwenhuizen@wur.nl](mailto:ard.Nieuwenhuizen@wur.nl)

Janne Kool

[janne.kool@wur.nl](mailto:janne.kool@wur.nl)

