

Runa Overå Hide

# Validation study of a video-based motion tracking with convolutional neural network for the take-off phase in ski jumping

Master's thesis in Physical activity and Health, with specialization in Movement Science

Supervisor: Espen Alexander F. Ihlen

May 2021



Runa Overå Hide

# **Validation study of a video-based motion tracking with convolutional neural network for the take-off phase in ski jumping**

Master's thesis in Physical activity and Health, with specialization in Movement Science

Supervisor: Espen Alexander F. Ihlen

May 2021

Norwegian University of Science and Technology

Faculty of Medicine and Health Sciences

Department of Neuromedicine and Movement Science



Norwegian University of  
Science and Technology



# VALIDATION STUDY OF EFFICIENTHOURGLASS IN SKI JUMPING

A new tool in analysing kinematics in sports

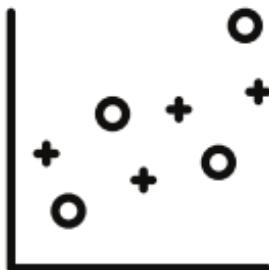


## EfficientHourglass?

- A type of convolutional neural network (CNN)
- Inspired by the brain and its connections
- The model automatically learns features in the image, and thus predicts the body key points of the ski jumper
- Seven different human raters have annotated 16 different body key points in 9324 images
- The human annotation is the "gold standard" for the EfficientHourglass

## Why is this interesting?

- Today, technique is measured by IMUs or manual video annotations, which requires several sensors to obtain precise results, it is subjective and time-consuming
- The EfficientHourglass is more time-efficient and give objective feedback regarding technique
- Few studies have stated the inter-rater error
- A need for an accessible tool to provide coaches and athletes to analyze technique during an in-hill jump



## What did we find?

- Achieved human inter-rater precision in two of the general performance metrics, PCKh
- The calculation of the the hip-, knee-, and ankle joint angles performed by the models were within the limit of error
- A tendency of increased precision as the network goes from -B0 to -B1, and from the small to the large network

*PCK : percentage of correct keypoints. "The fraction of detected keypoints that fall within a defined distance of the ground truth" (Mathis et. al, 2020).*

## Conclusion

- EfficientHourglass obtained the human inter-rater precision in several of the performance metrics
- Improvement of the human annotation will improve the model precision, as well as enable exclusive use of the visible side and post-processing
- The suggested methods above could result in improved precision



For more information see the study "Validation study of a video-based motion tracking with convolutional neural network for the take-off phase in ski jumps" (2021)

## Abstract

### Background

In this study, a type of convolutional neural network (CNN), EfficientHourglass, was validated analysing ski jumping technique. CNN has shown to be the state-of-the-art (SOTA) algorithm to solve challenging human pose estimation (HPE) and motion tracking tasks from video and images. Today, kinematic variables are obtained from 3D motion capture and from in-hill training and -competition jumps, IMUs, or manual video annotation. However, IMUs require several sensors to obtain precise results, and manual annotation are prone to subjective error. Thus, there is a lack of methods for in-competition analysis of the ski jump kinematics.

### Hypotheses

Two hypotheses were tested. That the EfficientHourglass was able 1) to detect the ski jumper body key points and 2) to identify hip-, knee-, and ankle joint angles, both with human expert precision.

### Methods

A dataset containing 9324 images of ski jumpers in the sagittal plane were annotated by 7 raters. Due to the size of the dataset, transfer learning and pretrained blocks on MPII were used in the encoder part on the CNN. Human inter-rater precision was calculated using 99 randomly chosen images from the dataset. The dataset was split into three subsets: training (72%), validation (8%) and test (20%). The method includes a description of the blocks included in the EfficientHourglass architecture.

### Results

All four models obtained the human precision of 90.86% in  $PCK_h@30$ . None obtained the human precision in  $PCK_h@10$  or  $PCK_h@error\_head$  of 52.7% and 0.1336, respectively. Noteworthy, top head, thorax, pelvis, right and left hip obtained low precision. The inflection point of the optimal image resolution in terms of precision against GLOPs was approximately 256x256 to 288x288 in all performance measures. Calculated joint angles by the models of the hip, knee and ankle were between 2.34° and 4.57°.

### Conclusion

This study confirmed the hypotheses on some of the performance metrics, that EfficientHourglass was able to detect the body key points of the ski jumpers and to calculate the three joint angles within the limit of error. A markerless motion tracking would result in a more objective and time-efficient measure of the kinematic variables. To improve the precision of the network, the precision of the raters must be improved by, e.g., a more detailed description of the body key points and perhaps several annotations for a body segment prone to high error/low precision (e.g., the hip joint). This will benefit the calculation of the joint angles and reduce the ME in degrees.

**Keywords:** convolutional neural network; human pose estimation; markerless motion tracking; kinematics; ski jumping

## Abstrakt

### Bakgrunn

I denne studien ble en type nevralt nettverk (CNN), EfficientHourglass, validert i analysering av teknikk i skihopp. CNN har vist seg til å være den siste og beste (SOTA) algoritmen til å løse komplekse oppgaver ved menneskelig- og markørløs bevegelsesanalyse i video og bilder. Per i dag blir kinematiske variabler målt ved bruk av 3D bevegelsesanalyse fra imitasjonshopp eller hopp fra konkurranser, IMUs eller manuell video annotering. Problemet er at flere sensorer (IMUs) trengs for å oppnå et presist svar og manuell annotering er disponert for subjektiv error. Det er en etterspørsel etter metoder å bruke ved hopp i konkurranser for å analysere kinematiske variabler i skihopp.

### Hypoteser

To hypoteser ble testet i studien: At EfficientHourglass var i stand til 1) å annotere skihopperens anatomiske landemerker og 2) å identifisere hoftel-, kne-, og ankel-ledd vinkler, begge med menneskelig presisjon.

### Metode

Et datasett som inkluderte 9324 bilder av skihoppere i sagittalplanet og ble annotert av 7 ulike annotører. Grunnet størrelsen på datasettet ble transfer learning og implementering av pre-trente blokker på MPII brukt i encoder-delen. Menneskelig inter-rater presisjon ble regnet ut ved å bruke 99 tilfeldig utvalgte bilder fra datasettet. Bildene ble delt inn i tre grupper; trening (72%), validering (8%) og test (20%). Metoden i studien beskriver blokkene inkludert i oppbyggingen av EfficientHourglass.

### Resultat

Alle fire modeller oppnådde menneskelig presisjon av 90.86% i  $PCK_h@30$ . Ingen nådde menneskelig presisjon av 52.7% i  $PCK_h@10$  eller ME av 0.1336 i  $PCK_h@error\_head$ . Høyre og venstre hoftel-, toppen av hodet, pelvis og thorax hadde lavest presisjon. Optimal bildeoppløsning ble satt til 288x288 med tanke på presisjon mot bruk av GLOPs i alle presisjonsmålinger. Utretnete leddvinkler av hoftel-, kne-, og ankel var mellom 2.34° og 4.57°.

### Konklusjon

Studien bekrefter hypotesene på noen av presisjonsmålingene, at EfficientHourglass var i stand til å annotere anatomiske landemerker på skihopperen og regne ut de tre leddvinklene innenfor en gitt grense for error. En markørløs bevegelsesanalyse vil resultere i en mer objektiv og tidseffektiv måling av kinematiske variabler. For å forbedre presisjonen av modellene, må presisjonen av menneskelig annotasjon forbedres, for eksempel ved en mer detaljert beskrivelse av annotasjonspunktene og ved å bruke gjennomsnittet av flere markører på et spesifikt punkt som per nå er utsatt for lav presisjon/høy error, som for eksempel hoftel-leddet. Dette vil påvirke utregningen av leddvinkler positivt og redusere ME i grader.

**Nøkkelord:** nevralt nettverk; markørløs bevegelsesanalyse; estimering av kroppslig posisjon; kinematiske variabler; skihopp

## Preface

What a year it has been! Both in terms of the current situation and writing this master thesis. The reason I chose this project is because of my interest in technology in a health and sports context, and it has always been appealing to me to be a part of something new. Though the first six months were a bit frustrating as the field is huge and something I have never used previously. After reading article on article about the different architectures and how it can be applied in many different contexts, the interest has increased and even though it has been hard, I am very happy for all the work I have put in the thesis.

First off, I would like to thank Espen for being a great supervisor and helping me throughout this year. Thanks for answering all my questions and for giving me lots of constructive feedback. Also, thanks to the rest of the project group, Daniel, Steinar, Gertjan and Ola.

Despite the pandemic, me and my fellow students have met at school and this has helped me tremendously. It feels a bit strange to say, but it sure has been a great source of energy and motivation throughout the day. In general, a HUGE thanks to my family, roomie Trine, and friends for giving me good energy (and pep-talks in times of need).

I do not think I have had a steeper learning curve throughout my years of higher education, and I hope that this master thesis can wake an interest in other people regarding human pose estimation and the appliance of CNN.

Best regards,

Runa





# 1. Introduction

For athletes competing in ski jumping, the result is mainly determined by the length of the jump. It is therefore important to have information regarding the technique and how to exploit the aerodynamics to perform optimally. The ski jump can be divided into four different phases: in-run, take-off, flight, and landing. All phases affect the jump, but early flight and take-off are considered to be the most crucial (1, 2). The main purpose of the in-run is to gain speed and establish optimal body position for the take-off phase (1, 3). The in-run is characterized by the ski jumper standing in a squat position, which needs to be low enough to reduce frontal drag and maximize horizontal speed (1), typically a 113 to 116 degree, 58 to 66 degree and 49 to 54 degree flexion in the hip, knee and ankle, respectively (4, 5). At the same time, the squat must be high enough to allow rapid extension of the lower extremities during take-off (1). A lower in-run position decreases the duration of take-off and increases the rate of force development (1). This will benefit the following stage, the take-off, which establishes the initial conditions for the flight. The take-off phase is often performed within 300 milliseconds and at speeds up to 25 m/s (6). The initial take-off phase is characterized by a 29 to 36 degree, 70 to 79 degree and 50 to 53 degree flexion in the hip, knee and ankle respectively (2, 7). The rate of force development is crucial for the take-off raising the center of mass (CoM) by a rapid knee and hip extension with typical angular acceleration of  $-92$  to  $-131^\circ/\text{s}$ , depending on the performance level of the ski jumper (1). This achievement produces forward-rotating angular momentum, which is needed to compensate for the backward-rotating angular momentum produced by drag on the skis during early flight (1). Previous studies have stated a positive correlation between acceleration of leg extension and length of the ski jump in the sagittal plane and between the position of the squat and length of the jump (1, 3, 7). Thus, investigating these kinematic variables, hip, knee, and ankle joint angles are important to evaluate the performance of a ski jump.

Due the challenges to practice multiple in-hill jumps within the same session, the hip-, knee-, and ankle joint angles are mostly assessed by in-lab imitation jumps (3, 8). However, the in-lab imitation jumps have several important differences compared to an in-hill jump: Firstly, the influence of air resistance is different in the two jump conditions. Virmavirta, Kivekas & Komi (2001) investigated what a difference in air resistance made, and found a 14% reduction in take-off duration (9). The lack of air resistance and friction between the track and the skis affects the in-run velocity (8, 9). Secondly, possibly due to the same reason, Schwameder & Müller (2001) observed a more apparent forward-oriented movement and higher take-off forces for the in-lab jumps compared to in-hill jumps. Thirdly, different types of shoes are used during in-lab practice and in-hill jumps. The shoes used in competition are stiff and prevent the foot from doing a plantar flexion, which minimizes air friction and maximizes lift (5, 8, 10). As the boundary conditions change, one would think the kinematic outcome would change, but it is still unknown to which degree (11). Together, these observations make it difficult to generalize results from studies of in-lab simulation jumps to actual in-hill competition jumps.

To analyze the kinematics of in-hill ski jumps different technologies have been utilized. The most familiar methods are recordings from inertial measurement units (IMUs) and video recordings (3, 10). IMUs are sensors containing accelerometers and gyroscopes which measure acceleration and change in orientation (i.e., rotation) for the body segment the IMU is placed on (12). IMUs were used by Logar & Munih (2015) to estimate joint forces and moments of six jumpers during the in-run and take-off. Two were attached to the skis in front of bindings, six were attached to shanks, thighs, upper arms and the two last at

the sacrum (13). Chardonens, Favre, Cuendet, Gremion & Aminian (2014) also used IMUs in analysis of 22 athletes, where five were placed on the body (sacrum, thighs and shanks), and two placed on the back of the skis (14). The IMUs are non-invasive and therefore convenient, and do not rely on external sources nor affected by external factors such as light conditions (12). However, the IMUs rely on double integration of the acceleration signal to calculate position, and thus prone to amplification of small drift in the original signal (12). Any minor error in data will cause possible result in error of positional estimates (12). For the IMUs to obtain accurate precision the sensor needs to be placed on correct anatomical landmarks and several sensors are required (15, 16). This may affect the ski jumper's performance negatively and decrease the reliability of the precision obtained by the IMUs (15). Consequently, most coaches of the athletes utilize video recordings to evaluate the kinematic variables, such as the hip-, knee-, and ankle joint angles.

Studies which have utilized manual annotation of video sequences or images are summarized in Table 1.

TABLE 1: A SUMMARY OF DIFFERENT STUDIES WHICH HAVE PERFORMED MANUAL VIDEO ANNOTATION, THE NUMBER OF JUMPS INCLUDED, WHICH BODY KEY POINTS WERE ANNOTATED, WHICH JOINT WERE INCLUDED, IF MEASURE OF INTER-RATER ERROR WAS INCLUDED AND WHICH PHASE OF THE SKI JUMP WAS INVESTIGATED.

Study	# of jumps	Equipment	2-D annotation	model	Angles utilized	Inter-rater error	Phase analyzed
Virmavirta, Isolehto, Komi, Schwameder, Pigozzi & Massazza (2009)	28	2 high-speed cameras (200 Hz). Stationary	7 unilateral segments (joint centers) calculated from manually digitized data		Upper body, hip, knee, shank	Not described in the study	Take-off
Arndt, Brüggemann, Virmavarta & Komi (1995)	20	2 3-CCD video cameras. 50 fields/s. NAC high speed video system (HVS400)	Four arm segments, six leg segments, torso, one head/neck		Torso, hip, knee, shank, somersault angle, COM-ankle	Not described in the study	Take-off and early flight
Lorenzetti, Ammann, Windmuller, Haberle, Müller, Gross, Plüss, Plüss, Schödler & Hübner (2019)	50	2 video cameras. 1 Legria HF R66 (50 Hz) for frontal plane, 1 Bosch for sagittal (50 Hz)	Neck/head, shoulder, hip, ankle	torso, knee,	Lower body angle, upper body angle, shoulder, hip, knee, ankle	Not described in the study	In-run and take-off
Virmavarta, Isolehto, Komi, Brüggemann, Müller, Schwameder (2005)	22	2 high-speed cameras (HSC-200). 200 frames/s.	12 segments. Undefined in article		Ski angle, body angle, upper body, angle, COM	Not described in the study	Early flight
Janurova, Janura, Cabell, Svoboda, Vareka, Elfmark (2013)	28	1 stationary camera. Grundig S-HVS 180 or Sony DCR-TRV 900, sampling frequency 50 Hz.	Shoulder, elbow, hip, knee, ankle		Body COM in sagittal plane, shoulder, elbow, hip, knee, ankle	Not described in the study	In-run
Janura, Cabell, Elfmark & Vaverka (2010)	15	1 stationary camera. Grundig S-HVS 180 or Sony DCR-TRV 900, sampling frequency 50 Hz.	Head, neck, upper arm, forearm (wrist included), trunk, thigh, shank, foot		COM angle, trunk, hip, knee, ankle	Not described in the study	In-run

Manual annotation provides trainers and athletes with useful information regarding the kinematics of the performed ski jump and, consequently, guidelines to further improve the technique. Even when performed by experts and trained analysts, manual annotation in sports can have limitations (17). Two challenges in manual video annotation are subjectiveness and the time-consuming work. Thus, it is only applied to a limited number of videos (15, 17). The task is often monotone and can eventually make the rater inconsistent in their annotation over time. This is called the "speed-accuracy trade-off", meaning the faster the annotation, the less precise and thus, higher inter-rater error (18). As most studies with manual annotation do not state the inter-rater error, it is difficult to determine the accuracy of the annotations.

Recently, to meet these challenges, innovative automated video-based motion tracking systems has been developed based on convolutional neural networks (CNN). It is a type of machine learning algorithm able to detect human skeletal key points from a sequence of video frames. It has shown to be the state-of-the-art (SOTA) algorithm to solve challenging human pose estimation (HPE) and motion tracking tasks of large-scale data sets such as MPII and COCO (19, 20). Different CNN architectures, such as OpenPose (21, 22), DeeperCut (23) and Stacked Hourglass (24), have been suggested as SOTA for HPE benchmarks. OpenPose was the first real-time multi-person system to jointly detect human body, hand, facial and foot key points on single images. OpenPose is a bottom-up approach, meaning it detects body key points for every person in the picture, followed by assigning parts to distinct individuals (21). DeeperCut is another multi-person pose system which uses the bottom-up approach. DeeperCut included convolutional layers in spatial models, which improved the overall accuracy of HPE (23). Different benchmarks of CNN have been applied in specific video-based motion tracking in sports like basketball, ballet, and tennis (25-27). These applications improved the precision compared to previous architectures, yet only Chen & Wang (2020) stated which architecture (LSTM) was used (25-27). Many of these SOTA CNNs have been used based on training on MPII and COCO without specific adaption to the task at hand. The focus in current SOTA CNNs seems to be designed for multi-person pose estimation with random occlusion e.g., body parts from other people. Thereby, the CNNs are often unnecessary complex and, consequently, more computer inefficient in appliance of simpler single-person HPE tasks such as kinematic analysis of a ski jumper in the sagittal plane.

Recently, EfficientHourglass was presented by Groos, Ramampiaro & Ihlen (2020) which outperforms other widely used CNN models, like OpenPose, in accuracy, size, and computational efficiency EfficientHourglass had a percentage of correct key points of 81.2% compared to 34.7% for OpenPose (28). The EfficientHourglass had 1.4-54x fewer parameters and a 2.2-168x reduction in number of floating operations (FLOPs) and an overall 16x speed-up of inference was achieved (28). Thus, EfficientHourglass may be a suitable CNN for automatic video-based motion tracking of ski jumpers in the take-off phase and for the kinematic assessment of the ski jumper's hip-, knee-, and ankle joint angles in the take-off phase.

The research aim of the thesis is to validate the EfficientHourglass CNN in markerless tracking of ski jump kinematics: hip-, knee-, and ankle joint angles during take-off. There are two hypotheses that will be tested: That the EfficientHourglass is able to 1) detect the ski jumper body key points and 2) identify hip-, knee-, and ankle joint angles, both hypotheses with human expert precision.

## 2. Method

In this section the ski jumper pose data set and the EfficientHourglass architecture are presented, including the pre-processing, training procedure and performance metrics utilized to evaluate the precision of the different network models.

### 2.1 Ski jumper pose dataset

The ski jumper dataset contained a total of 9324 images with 149184 body key points labels collected from ~4 images per video of ski jumpers. The ski jumps were collected from 41 different in-hill training- or competition jumps, both female and male elite athletes on either national or international level. The 16 body key points were compromised to a full body kinematic model of the ski jumper: top of the head, upper neck, shoulders, elbows, wrists, upper chest, right/mid/left pelvis, knees, and ankles (see Figure 2).

#### 2.1.1 Manual annotation

The manual annotation was performed by 7 raters. One works at Granåsen Toppidrettssenter, five have a background in Movement Science and one has a background in Computer Science. Each rater was given a full description of the procedure regarding the manual annotation. The body key points were described and depicted as shown in Figure 1. Prior to the manual annotation, each had to correctly complete a practice set of five images of ski jumpers to ensure that the rater had a correct interpretation of the body key point description. The full description each rater was handed prior to the task is available in Appendix 1.

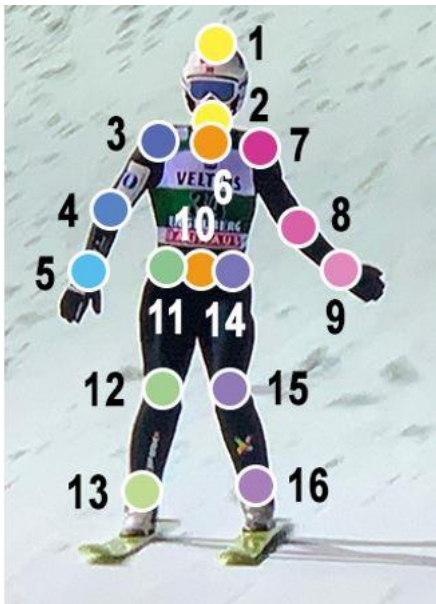


FIGURE 1: PICTURE GUIDELINE FOR HOW THE BODY KEY POINTS SHOULD BE ANNOTATED.

#### 2.1.2 Inter-rater error

From the 9324 images, 100 images were randomly chosen to be annotated by each rater. One of the images was excluded due to difficulties in annotation (n=99). The calculation of the inter-rater error was performed to ensure the degree of agreement among raters, and to affirm the validity of the human annotation. The inter-rater error is the “gold standard” for the automated annotation by the CNN.

## 2.2 Architecture

### 2.2.1 Top-down approach

EfficientHourglass is applied in a top-down approach where the ski jumper is first detected in the video frame and then EfficientHourglass is applied for HPE of the identified bounding-box of the ski jumper (Figure 2). In this study the bounding-box is created with a frame with size 7.5-15% of the distance range of the annotated body key points. The frame size was randomized in the range 5-10% to conceal the exact position of the body key points. The bounding box image was the input image for the EfficientHourglass networks.

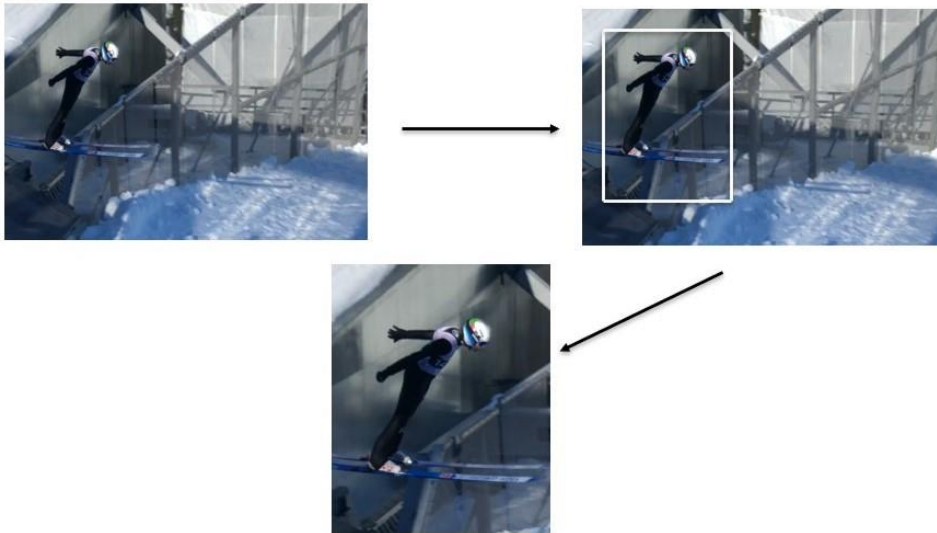


FIGURE 2: ILLUSTRATION OF HOW EFFICIENTHOURGLASS WORKS IN A TOP-DOWN APPROACH, INCLUDING APPLIANCE OF THE BOUNDING BOX.

### 2.2.2 EfficientHourglass architecture

The EfficientHourglass network contains two main parts: the encoder and the decoder part. The encoder part is the pretrained blocks from EfficientNet (29), which is illustrated as the blue blocks depicted in Figure 3. This part downscales the input image resolution and transforms the input pixels to features. A feature contains descriptive information, such as edges, lines or colour intensity (30). CNNs, and EfficientHourglass, learns simple edge detectors in the early layers and more abstract features in the deeper layers (31). The decoder part upsamples the image resolution from a low input image resolution to a high output image resolution (32), and illustrated as the green blocks in Figure 3. The EfficientHourglass network architecture is inspired by the single-stage hourglass architecture (24). Thus, information from different image resolutions were connected by bridge blocks containing feature maps from block 2, 3 and 5, depending on the size of the network.

### 2.2.3 Small and large network

To find an optimal complexity (number of parameters) of the architecture in terms of performance, a large and small network of EfficientHourglass were developed as described below.

Large network: Block 1 – 6 of EfficientNetB0 and -B1 was included, whilst block 7 was excluded (29). The result is a reduction to 1/32 of the input image resolution at the end of Block 6. Three transpose convolutions, including the bridge connections from block 2, 3 and 5, are performed to upscale the feature maps to the final output confidence map.

Small network: Block 1 – 5 of EfficientNetB0 and -B1 was included. Block 6 and 7 was excluded to reduce the complexity of the network. The result is 1/16 of the input image resolution. This was followed by two transpose convolutions with two bridge connections from block 2 and 3.

In total four models of EfficientHourglass were tested to investigate the influence of different width and depth on the ski jumping motion tracking performance. The architecture is presented in Table 2.

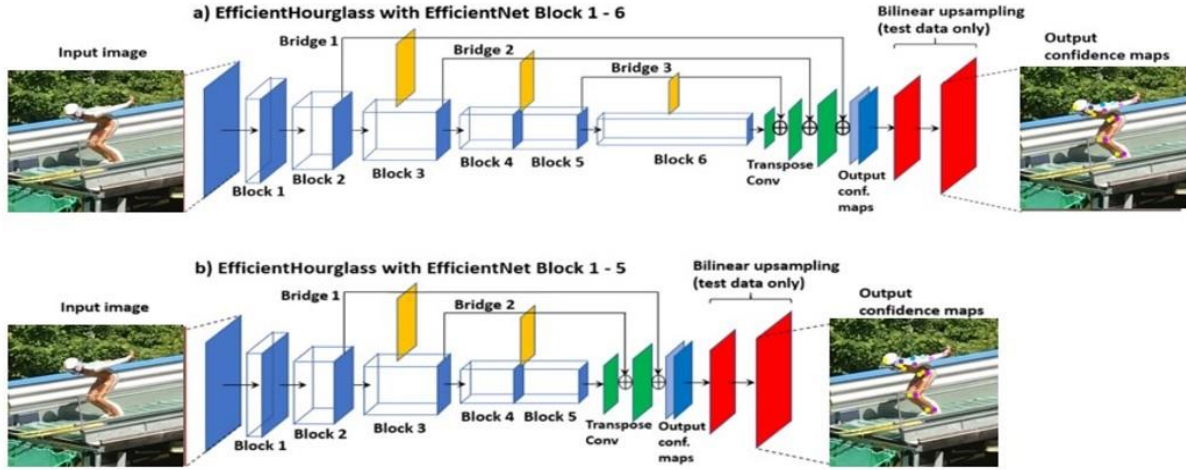


FIGURE 3: A) SHOWS THE DIFFERENT STAGES IN THE LARGE NETWORK, AND B) FOR THE SMALL NETWORK. THE BLUE BOXES ARE THE ENCODER PART OF THE NETWORKS. THE TRANSPOSE CONVOLUTIONS ARE THE GREEN BLOCKS. OUTPUT CONFIDENCE MAPS COMBINE THE FEATURES FROM THE DIFFERENT FEATURE AMPS. THE OUTPUT IS AN ANNOTATED IMAGE OF A SKI JUMPER.

TABLE 2: HOW THE EFFICIENTHOURGLASSB0 AND -B1 ARE ORGANIZED, IN BOTH THE SMALL (BLOCK1TO5) AND LARGE NETWORK (BLOCK1TO6) FROM FIGURE 4. S = STRIDE CONVOLUTION.

Block	Layer	Output size	B0 block1to5	B1 block1to5	B0 block1to6	B1 block1to6
→ 1 1	Conv, S MBConv1	1/2	[3x3, 32] [3x3, 16] x1	[3x3, 32] [3x3, 16] x2	[3x3, 32] [3x3, 16] x1	[3x3, 32] [3x3, 16] x2
1 → 2 2	MBConv6, S, MBConv6	1/4	[3x3, 24] [3x3, 24] x1	[3x3, 24] [3x3, 24] x2	[3x3, 24] [3x3, 24] x1	[3x3, 24] [3x3, 24] x2
2 → 3 3	MBConv6, S, MBConv6	1/8	[5x5, 40] [5x5, 40] x1	[5x5, 40] [5x5, 40] x2	[5x5, 40] [5x5, 40] x1	[5x5, 40] [5x5, 40] x2
3 → 4 4	MBConv6, S, MBConv6	1/16	[3x3, 80] [3x3, 80] x2	[3x3, 80] [3x3, 80] x3	[3x3, 80] [3x3, 80] x2	[3x3, 80] [3x3, 80] x3
5	MBConv6		[5x5, 112] x3	[5x5, 112] x4	[5x5, 112] x3	[5x5, 112] x4
5 → 6 6	MBConv6, S, MBConv6	1/32	----	----	[5x5, 192] [5x5, 192] x4	[5x5, 192] [5x5, 192] x4

The main building block of EfficientHourglass is the mobile inverted bottleneck sub-blocks (MBConv). The MBConv-block have the following layers:

1) A 1x1 convolution (conv) inverted bottleneck which increases the number of input channels from N to mN where m is a factor (see number within brackets in Table 2).

EfficientHourglass uses  $m=1$  and 6 defining MBConv1 and MBConv6. For  $m=1$ , the  $1 \times 1$  conv is omitted.

2) A depth-wise convolution (dConv) with a receptive field of  $3 \times 3$  or  $5 \times 5$  is the feature extractor in the MBConv-block and is more efficient compared to ordinary convolution in terms of FLOPs because the convolution is conducted channel-wise.

3) Squeeze-and-excitation (SE) layer provides channel-wise attention by assigning each channel (e.g., feature map) a value between 0 and 1, where 0 means no significance and 1 means great significance for body key point detection (33).

4) A  $1 \times 1$  convolution bottleneck decreases the number of channels of the feature tensor from  $mN$  to  $N$  before a residual connection adds the feature tensor of the former MBConv-block. The  $1 \times 1$  conv increases/reduces the number of channels in the feature tensor.

All convolution layers in the MBConv block are followed by a batch normalization (BN) layer that standardizes the means and variances of the input pixels to accelerate the training of the network (34). All BN layers are followed by a non-linear activation function, called Swish, for the network to learn complex features (35). See Appendix 2 for more detailed information regarding CNN layers like BN, activation function Swish, and SE-layer.

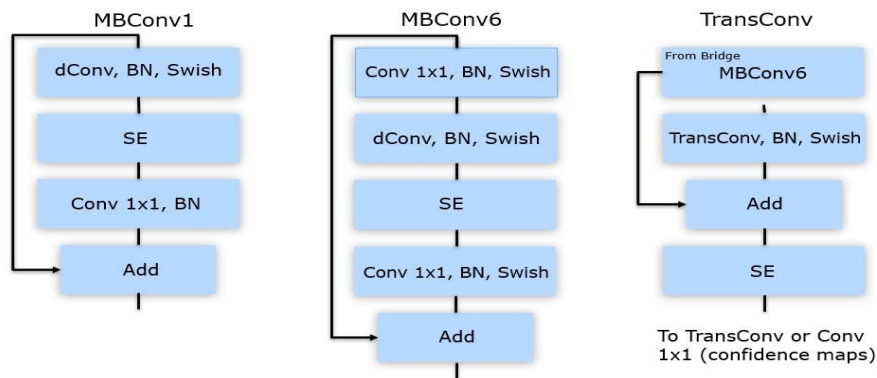


FIGURE 4: DESCRIPTION OF THE DIFFERENT BLOCKS FOUND IN THE NETWORK. MBCONV1 IS USED IN BLOCK 1, MBCONV6 IS USED IN BLOCK 2-6 IN THE LARGE NETWORK, OR 2-5 IN THE SMALL NETWORK.

## 2.3 Training of EfficientHourglass

The model architecture was developed in TensorFlow, and the training was performed on three GPUs (Nvidia GTX 1080 Ti, Nvidia RTX 3090 and Quadro RTX 8000).

### 2.3.1 Input resolution

The original image resolution of the 9324 annotated images was cropped and resized in a squared frame with image resolution  $1024 \times 1024$ . All four models of EfficientHourglass were tested on multiple resized input image resolutions ranging from  $128 \times 128$  to  $512 \times 512$ . This was performed to find the optimal image resolution in body key point detection.

### 2.3.2 Training-, validation- and test subset

The total dataset was divided into three different subsets: training, validation, and test. These included 6713 (72%), 747 (8%), and 1864 (20%) images, respectively. The training subset was used to fit the models' weights, whilst the validation dataset was used to evaluate the performance of the training. The final evaluation was performed on the test subset, and this subset was separated from the two others to ensure reliable results. Training and validation of the model was performed until the model converged to optimal



performance. The training-, validation-, and test subset was stratified for a person-hill combination: i.e., Lillehammer140\_Granerud, and Lillehammer140\_Tande. This to ensure that images from the same person-hill trial was included exclusively in only one of the subsets to prevent an over-optimistic performance.

### 2.3.3 Exclusion of images

After the raters had manually annotated the entire dataset, the images and their respective annotations were reviewed in MATLAB to ensure applicable annotations. If the annotations were placed in the corner or in the top of the image it would be excluded. Such errors could affect the data training and thereby the precision of the models negatively. This resulted in a total of 12 excluded images due to the ski jumper did not appear in the image (n=1) and only half of the ski jumper was visible (n=11).

### 2.3.4 Transfer learning

Due to the relatively small size of the dataset, transfer learning was used in training. Transfer learning is where pretrained weights for one task are fine-tuned to a related second task (30). To fine-tune the model, the pretrained weights in the network layers were used to learn the body features of the ski jumpers. Such fine-tuning of the pretrained network weights can reduce time of training and improve the overall precision of the model (30). The encoder part of the EfficientHourglass was pretrained on ImageNet, a large database of 14 million images organized in a hierarchical WordNet, meaning the images are described in words and sentences (36). The entire EfficientHourglass network was then pretrained on MPII Human Pose data base (19), for evaluation of articulated human pose estimation which contains ~25 000 images of different human activities.

### 2.3.5 Optimization of EfficientHourglass

The model automatically learns features from the images by continually updating the network weights during training. The loss function takes the prediction of a body key point made by the network and compares it to the ground truth, the human annotation (30). It is an evaluation of the models' performance on the training data. The loss function acts as a feedback to adjust the values of the input weights. The goal of an optimally trained model is achieved when the global minimum of the loss function is found. The optimizer of the model is the mechanism that will change the gradient of the loss function to update the weights. The gradient of the loss function and changes in weights gives information on the rate of the loss function, and thereby if the training is close to a minimum and how fast. The partial derivatives in each mini-batch are collectively called the gradient (31).

The dataset is divided into mini-batches with a batch size of 16 images, and the result is 419 mini-batches ( $6713/16=419$ ). After 419 iterations, the model has completed one epoch of training. Each iteration consists of a forward-pass (evaluation of the model) and a backpropagation (adjustment of the weights), as shown in Figure 6. After each epoch, the model is evaluated against the validation subset and the validation-loss is used as input in the next epoch. In EfficientHourglass, the number of epochs is set to 50. The learning rate can be seen as the magnitude of change of the model weights after each iteration to achieve a minimum of the loss function during training (31). The adjustments of weights are a continuous process, and the backpropagation (BP) algorithm uses the chain rule of calculus to compute the derivative (31).  $\text{New weight} \rightarrow (\text{old weights}) + (\text{learning rate}) * (\text{gradient})$ .

In EfficientHourglass, the Adam optimizer is utilized due to its adaptive learning rate which adjusts according to how close the loss function is to its minimum during training (37).

Data augmentation is used, and each image in a mini-batch is rotated +/- 45 degrees, rescaled +/- 0.25 times and flipped (right/left) to ensure variation of the images in the training subset and prevent overtraining of the model.

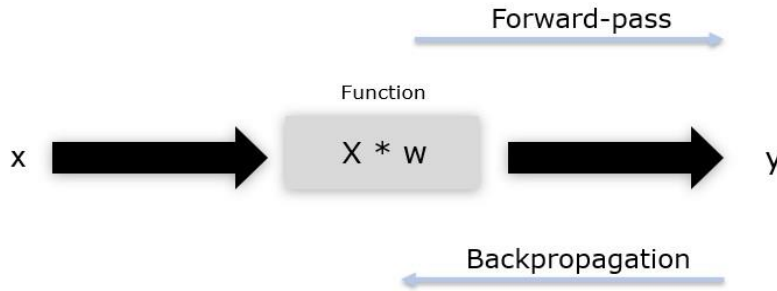


FIGURE 5: X IS THE INPUT, W IS THE WEIGHTS (IN EACH FILTER), AND Y IS THE OUTPUT AFTER THE FUNCTION  $x*w$ .

## 2.4 General performance metrics

To evaluate the EfficientHourglass models, the performance metrics described below were used.

### 2.4.1 The mean error relative to head size

The mean error of head-size (ME-h) defines the average precision on the body key points  $b$ , where the estimated body key points by the EfficientHourglass is  $[\bar{x}, \bar{y}]$  and by the raters  $[x, y]$ .

$$ME - h_b = \frac{1}{n} \sum_{i=1}^n \frac{d_{b,i}}{l_i} \quad (1)$$

Where  $l_i$  is the Euclidian distance between the annotated position of key point of top head and upper neck.

The Euclidian distance  $d_{b,i}$  is compute for each key point  $b$  and image  $i$  is given by:

$$d_{b,i} = \sqrt{(x_{b,i} - \bar{x}_{b,i})^2 + (y_{b,i} - \bar{y}_{b,i})^2} \quad (2)$$

The average in Equation (2) is computed in each image  $n$  in the test set. ME-h is given as a relative distance between 0 and 1 across the diagonal of the image for each body key point in the general performance metrics. The ME-h is in percentage of the head size of the ski jumper, e.g., a ME-h of 0.133 = 13% of the head size, where 1.0 = 100% of the head size. Using a relative error, like ME-h and  $PCK_h$  below, will make it easier to interpret results across different studies irrespective of the chosen video frame pre-processing.

### 2.4.2 $PCK_h@T$

PCK stands for percentage of correct key points. The equation includes a percentage  $\tau$  of the ski jumpers head size.

$$PCKh@T = \frac{\sum_{i=1}^n \delta(d_{b,i} < \frac{\tau}{100} l_i)}{n} * 100\% \quad (3)$$

$\delta$  is the Boolean operator that are equal to 1 when the argument is satisfied and 0 otherwise and where distance  $d_{b,i}$  is given by Equation (2). The smaller  $\tau$ , the smaller error area for the spesific body key point to be placed within. In this study the  $\tau$  was calculated to be 10% and 30% of the head segment size  $l_i$ , the Euclidian distance between upper neck and

top head key point. E.g.,  $PCK_h@10$  means that the annotated body key point needs to be within that circle that marks 10% of the head segment size (Figure 4).



FIGURE 6: THE BLUE CIRCLE IS  $PCK_h@10$ , THE RED CIRCLE IS  $PCK_h@30$  AND THE YELLOW CIRCLE IS  $PCK_h@50$ . THE TWO FIRST WILL BE USED IN THE STUDY, AND THE LAST IS MENTIONED AS IT IS THE ACTUAL SIZE OF THE HEAD.

### 2.4.3 Inter-rater spread

$ME-h$  and  $PCK_h$  were also calculated for the human annotation, for a comparison of the models to the human inter-rater precision (HIRP). The HIRP is based on the 99 similar images the 7 raters annotated prior to training where the Euclidian distance in Equation (3) is between the individual rater and inter-rater mean.

### 2.4.4 Illustration of results

The general performance metrics were computed for each of the 16 body key points and as a mean value across all points. The mean values of the performance metrics were presented against the computer efficiency of the network, giga floating operations per second (GLOPs), for all four models for different input image resolutions from 128x128 to 512x512. GLOPs show the efficiency of the network, mainly related to the network design and the specifically used GPUs (38).

## 2.5 Ski jump specific metrics

The second part of the hypothesis was to see if the EfficientHourglass was able to calculate the hip-, knee-, and ankle joint angles during the take-off phase.

The annotation of body key points resulted in x- and y-coordinates assessed in a csv-file. All calculations used inverse tangent conversion. The ski jump specific metrics are presented as mean error (ME) in degrees against GLOPs for all four models for different input image resolutions from 128x128 to 512x512.

### 2.5.1 Hip-, knee-, and ankle joint angle calculation

As an example, equation (6), (7) and (8) show how the hip was calculated the annotations from the ground truth:

$$Xa_{hip} = [Upper_{body}y_{x\{ang\}} - Hip\_x\{ang\}, 0] \quad (6)$$

$$Xb_{hip} = [Knee\_x\{ang\} - Hip\_x\{ang\}, 0] \quad (7)$$

$$Xc_{hip} = cross(Xa_{hip}, Xb_{hip}) \quad (8)$$

The hip joint was calculated using the upper body (the mean of the thorax, right shoulder and left shoulder markers) and the hip (the mean of the right hip, left hip and pelvis markers) and the knee (the mean of the right and left knee markers). See Figure 7.

The knee joint was calculated using the hip (from above) and ankle (the mean of the right and left ankle markers).

As the ankle marker was the last marker in the annotation, the ankle (the mean of both ankles) and the x-coordinates from the knee annotation (the horizontal axis) were used.

As not all triangles were right-angled triangles, the given equation was used to calculate the models ground truth in MATLAB:

$$x_{angle} = 180/\pi * atan2(norm(Xc_{angle}), dot(Xa_{angle}, Xb_{angle})) \quad (4)$$

A similar equation was used to calculate the models' joint angles:

$$y_{angle} = 180/\pi * atan2(norm(Yc_{angle}), dot(Ya_{angle}, Yb_{angle})) \quad (5)$$

where  $x_{theta\_angle}$  or  $y_{theta\_angle}$  is the unknown joint angle,  $180/\pi$  to get the answer in degrees,  $atan2$  (inverse tangent),  $X_c$  and  $y_c$  is the cross product of the two known vectors in the angle (third is the z-direction which is unknown and set equal to 0).  $X_a$  and  $y_a$  use the x- and y-coordinates in two of the known body key points of the angle.  $X_b$  and  $y_b$  use the x- and y-coordinates of the two other known body key points used in the angle.

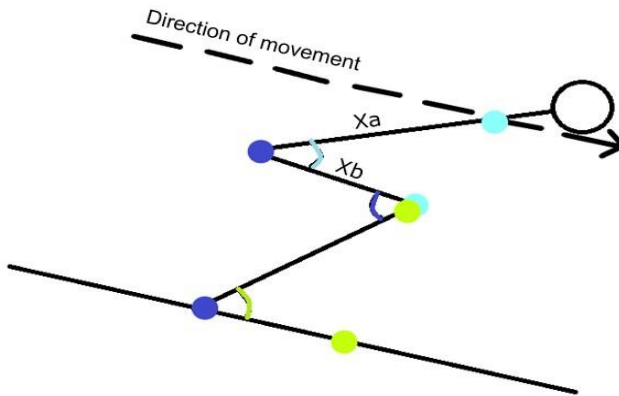


FIGURE 7: THE LIGHT BLUE COLOURS ILLUSTRATE CALCULATION OF THE HIP ANGLE USING THE UPPER BODY/THORAX (MEAN ACROSS UPPER BODY, RIGHT SHOULDER AND LEFT SHOULDER) AND KNEES; THE DARK BLUE COLOURS ILLUSTRATE THE KNEE ANGLE USING THE HIP (MEAN OF PELVIS, RIGHT AND LEFT HIP) AND ANKLES; THE GREEN ILLUSTRATES THE ANKLE ANGLE USING THE KNEES AND HORIZONTAL POINT OF THE KNEE ANNOTATION.

### 3. Results

As an illustration of how the models annotated the body key points, three annotated images from the test subset are presented in Figure 8. Two of the images are annotated correctly, but the third has misplaced some of the annotations. None of the four models achieved human inter-rater precision (HIRP) in the  $PCK_h@10$  (Figure 10 (A)). All four models obtained a higher percentage than HIRP in the  $PCK_h@30$  (Figure 10 (B)) where -B1 block1to5 obtained the highest precision. In the  $PCK_h@error\_head$  in Figure 11, HIRP obtained a ME-h of 0.1336, where -B1 block1to6 and -B0 block1to6 were close. All models, except -B0 block1to5 in the knee- and ankle joint angles, were able to calculate joint angles in accordance with the ME in degrees obtained by the HIRP (Figure 12). The inflection point of the performance measure graph plotted against the number of floating operations indicate the optimal resolution for each of the models. This was found to be at 256x256 to 288x288 for most of the performance measures, see Figure 9 and 12.

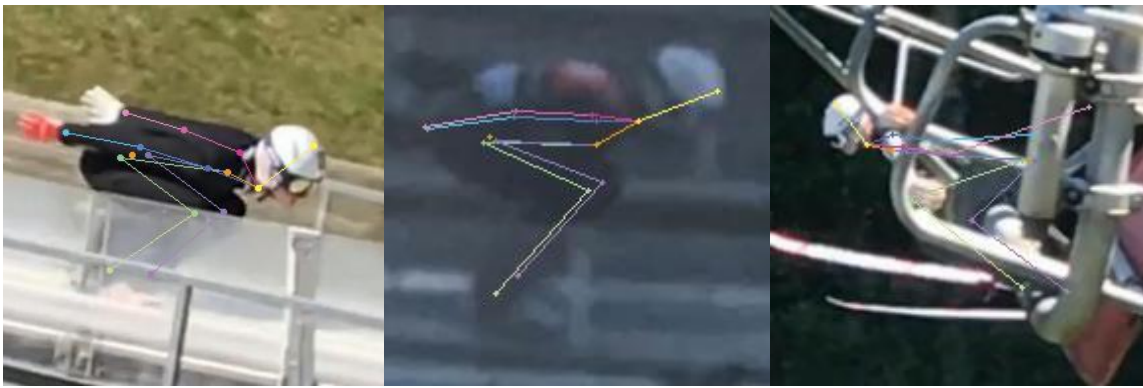
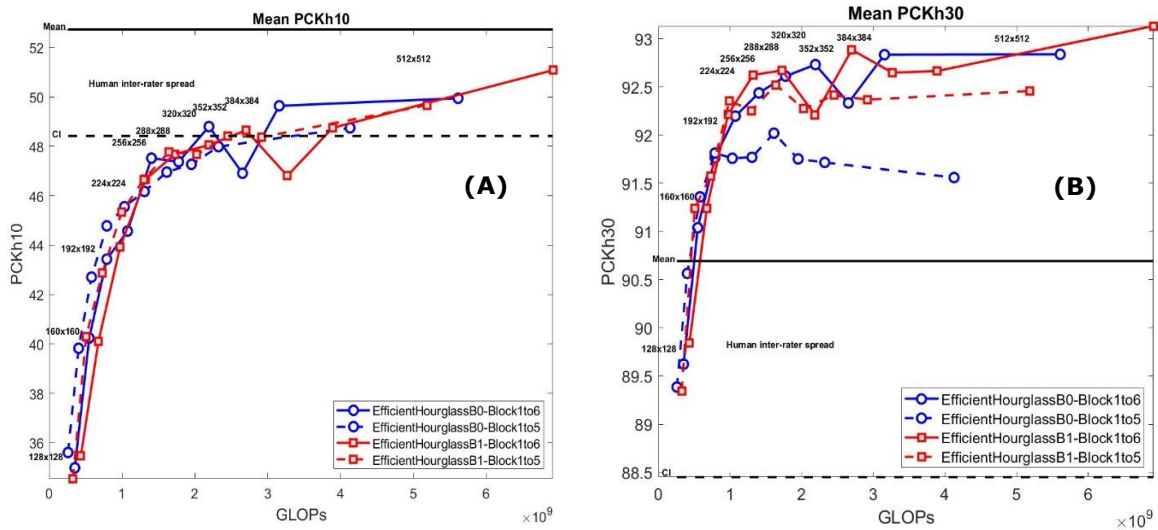


FIGURE 8: THREE EXAMPLES ON HOW THE MODELS HAVE ANNOTATED THE BODY KEY POINTS. THE LEFT AND MIDDLE IMAGE IS EFFICIENTHOURGLASSB1 BLOCK1TO6 IN IMAGE RESOLUTION 192X192. THE LEFT IMAGE IS EFFICIENTHOURGLASSB0 BLOCK1TO5 IN 192X192.

#### 3.1 General performance metrics

The mean precision or ME-h in different image resolutions of  $PCK_h@10$  (A),  $PCK_h@30$  (B) and  $PCK_h@error\_head$  (C) are presented in Figure 9, 10 and 11 and Table 3.



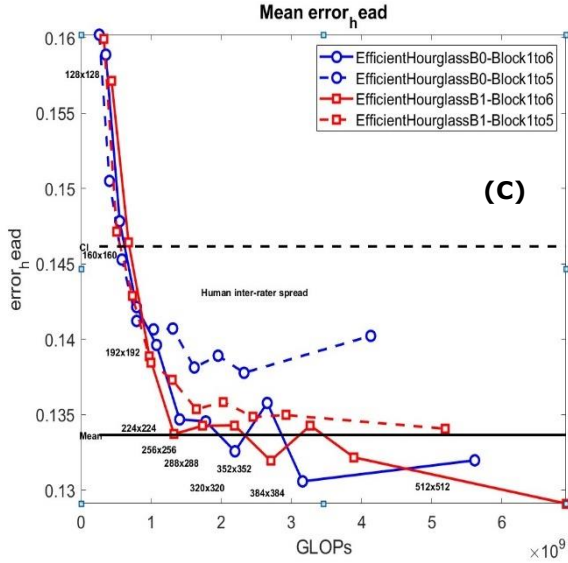


FIGURE 9: (A) MEAN PRECISION AT PCKH@10, (B) MEAN PRECISION AT PCKH@30 AND (C) MEAN PRECISION MEASURED IN ME-H FOR THE PCKH@ERROR\_HEAD. THE Y-AXIS: PRECISION IN PERCENTAGE, OR ME-H. THE X-AXIS: NUMBER OF GLOPs. THE BLACK DOTTED LINE: CONFIDENCE INTERVAL; THE BLACK LINE: HUMAN INTER-RATER PRECISION (HIRP).

Seen in Figure 9, the HIRP obtained a precision of 52.7% in PCK<sub>h</sub>@10, 92.72% in the PCK<sub>h</sub>@30 and 0.1336 in PCK<sub>h</sub>@error\_head. The models obtained a precision above the black line illustrating HIRP in Figure 9 (B) but were under in Figure 9 (A). Some of the models obtained HIRP in Figure 9 (C), but this is from an image resolution of 320x320 and increasing. In Figure 9, the models are observed to reach a plateau and flatten out around 288x288 and 320x320. This is due to the

models obtaining a close precision to the HIRP, which is their limit of performance. The only model clearly obtaining a lower precision than the HIRP is -B0 block1to5. The best performing model seems to be -B1 block1to5 or -B1 block1to6, but the -B0 block1to6 is also close in precision and GLOPs (Table 3). The model precision increases as image resolution increases, but a higher image resolution does not equal better performance and will to a large extent be dependent on the precision of the raters. A larger image resolution, e.g., in Figure 9 between 384x384 and 288x288, doubles the number of GLOPs for a ~1 percentage difference and a reduction of ~0.04 in ME-h, depending on the model. The inflection point seems to be around 256x256 and 288x288. Table 3 is an overview over the different performance metrics, the models' precision in each and use of GLOPs.

TABLE 3: @10, @30 AND @E\_H ARE ABBREVIATIONS FOR PCKH@10, PCKH@30 AND PCKH@ERROR\_HEAD, RESPECTIVELY.

<b>EfficientHourglass</b>	<b>Image resolution</b>	<b>@10</b>	<b>@30</b>	<b>@e_h</b>	<b>GLOPs</b>
<b>-B0 block1to5</b>	256x256	45.8%	91.8%	0.1407	1.032
<b>-B0 block1to5</b>	288x288	46%	91.8%	0.1407	1.306
<b>-B0 block1to6</b>	256x256	47.7%	92.4%	0.1347	1.404
<b>-B0 block1to6</b>	288x288	47.6%	92.7%	0.1344	1.775
<b>-B1 block1to5</b>	256x256	46.6%	92.3%	0.1373	1.297
<b>-B1 block1to5</b>	288x288	47.8%	92.5%	0.1353	1.641
<b>-B1 block1to6</b>	256x256	47.8%	92.8%	0.1343	1.728
<b>-B1 block1to6</b>	288x288	49.1%	92.2%	0.1343	2.188

The models obtained the closest precision to the HIRP in 288x288, especially -B1 block1to5 and -B1 block1to6 (Table 3). Also, the use of GLOPs in that image resolution did not differ that much between the models. In PCK<sub>h</sub>@error\_head, two of the models were particularly close, -B0 block1to6 and -B1 block1to6. These obtained a ME-h of 0.1344 and 0.1343, respectively. The small network (-B0) used 1.775 GLOPs whilst the large network (-B1) used 2.188 GLOPs. Thus, the image resolution 288x288 will be used as an example to present the obtained percentage of each body key point.

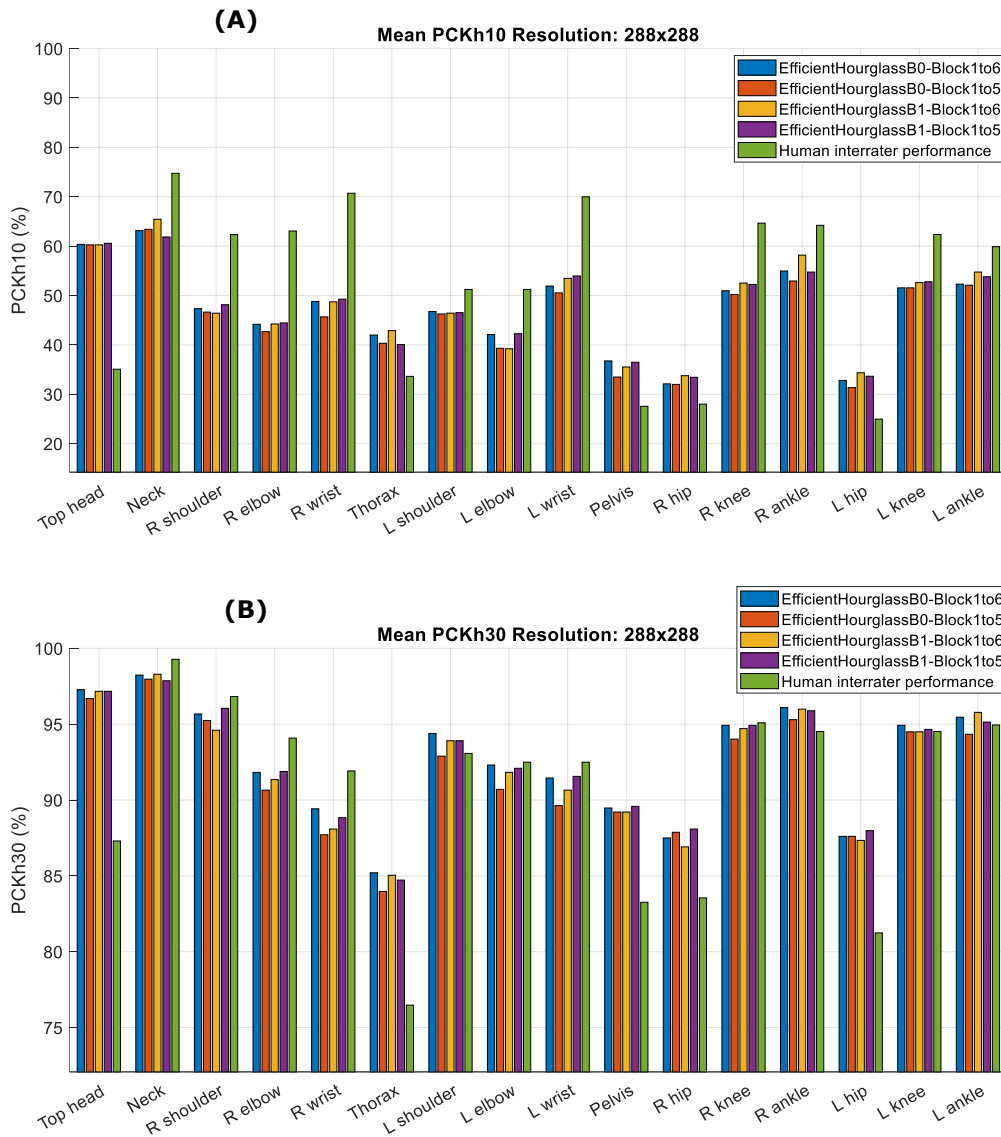


FIGURE 10: THE X-AXIS SHOWS THE 16 BODY KEY POINTS OF THE SKI JUMPER, AND THE Y-AXIS SHOWS THE ACCURACY IN PERCENTAGE. (A) FOR PCKH@10 AND (B) FOR PCKH@30.

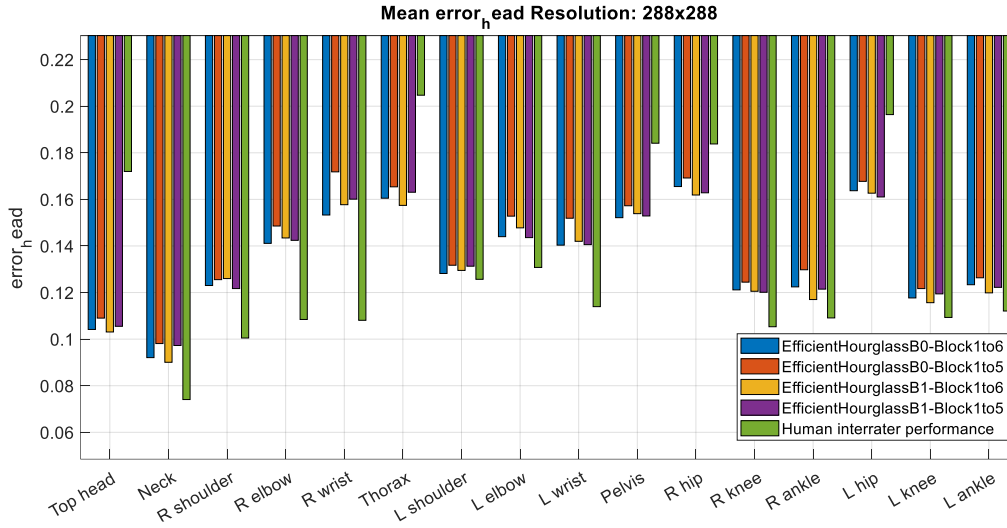


FIGURE 11: THE X-AXIS SHOWS THE 16 BODY KEY POINTS ANNOTATED, AND THE Y-AXIS SHOWS THE ME-H.

In Figure 10 (A), none of the models obtained the HIRP of 52.7%. The -B1 block1to6 had the highest precision with 48.94%, followed by -B1 block1to5, -B0 block1to6 and -B0 block1to5, respectively. Worth noticing, the annotation of the top head, pelvis, right hip, left hip, and thorax obtained a low precision. The HIRP obtained a precision of 27.56%, 27.99%, 24.96% and 33.62%, respectively. In -B1 block1to6, the same body key points obtained a precision of 60.26%, 35.52%, 33.72%, 34.35% and 42.9%, respectively.

Regarding Figure 10 (B), the HIRP obtained a mean precision of 90.69% compared to the best performing model, -B1 block1to5, which obtained a mean precision of 92.71%, followed by -B0 block1to6, -B1 block1to6 and -B0 block1to5. All four models achieved the HIRP in PCK<sub>h</sub>@30, and did especially well on right shoulder, right knee, left elbow, and left wrist. The -B1 block1to5 obtained a precision of 96.05%, 94.93%, 92.09% and 91.56%, respectively compared to the HIRP of 96.83%, 95.09%, 92.2% and 92.5% on the same body key points.

Seen in Figure 11, the HIRP obtained a ME-h of 0.1336. The tendency of the model precision is the same in this performance metrics, as the HIRP obtained a higher ME-h on the body key points top head, thorax, pelvis and right and left hip, and obtained a low ME-h on both shoulders and wrists. A ME-h of 0.172, 0.205, 0.181, 0.834 and 0.196, respectively. As -B0 block1to6 used less GLOPs than -B1 block1to6, with only a 0.001 difference in obtained ME-h, that model will be used as an example (Table 3). It obtained a ME-h of 0.106, 0.162, 0.155, 0.164 and 0.165, respectively.



### 3.2 Ski jump specific metrics

Figure 12 compares the predicted hip- (A), knee- (B), and ankle (C) joint angle of ME in degrees for the four models with increasing image resolution to HIRP.

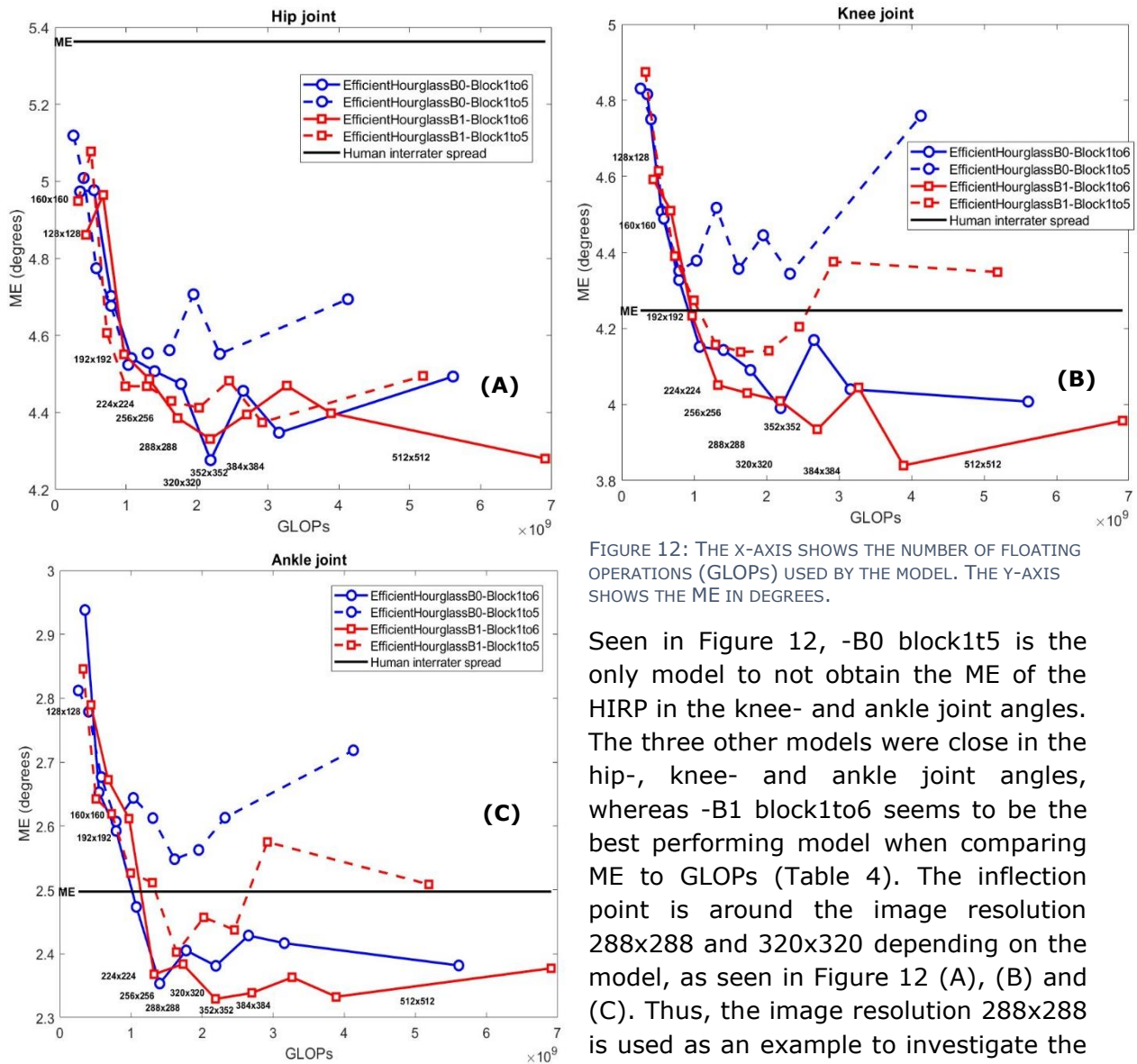


FIGURE 12: THE X-AXIS SHOWS THE NUMBER OF FLOATING OPERATIONS (GLOPs) USED BY THE MODEL. THE Y-AXIS SHOWS THE ME IN DEGREES.

Seen in Figure 12, -B0 block1t5 is the only model to not obtain the ME of the HIRP in the knee- and ankle joint angles. The three other models were close in the hip-, knee- and ankle joint angles, whereas -B1 block1to6 seems to be the best performing model when comparing ME to GLOPs (Table 4). The inflection point is around the image resolution 288x288 and 320x320 depending on the model, as seen in Figure 12 (A), (B) and (C). Thus, the image resolution 288x288 is used as an example to investigate the obtained ME in degrees between the four models.

The HIRP obtained a ME of 5.36°, 4.24° and 2.49° in the hip-, knee-, and ankle joint angle, respectively. The ankle joint has a generally lower ME, both in the HIRP and the four models. All ankle joints were calculated between a ME of 2.32° to 2.61°. The models obtained a ME between 4.31° to 4.55° in the hip joint angle, and 4.01° and 4.52° in the knee joint angle (Table 4).

TABLE 4: AN OVERVIEW OVER THE THREE CALCULATED JOINTS IN THE IMAGE RESOLUTION 288x288. AVERAGE ME IN DEGREES AND NUMBER OF GLOPS.

<b>Model</b>	<b>Ankle</b>	<b>Knee</b>	<b>Hip</b>	<b>GLOPs</b>
<b>-B0 block1to5</b>	2.61°	4.52°	4.55°	1.306
<b>-B0 block1to6</b>	2.38°	4.09°	4.47°	1.775
<b>-B1 block1to5</b>	2.40°	4.14°	4.43°	1.641
<b>-B1 block1to6</b>	2.32°	4.01°	4.31°	2.188
<b>Human inter-rater precision</b>	2.49°	4.24°	5.36°	-----

In summary, the models -B0 block1to6, -B1 block1to5 and -B1 block1to6 were able to detect the 16 body key points of the ski jumpers and the hip-, knee-, and ankle joint angles with HIRP.

## 4. Discussion

The study aimed to validate the EfficientHourglass CNN in markerless motion tracking of ski jump kinematics: hip-, knee-, and ankle joint angles during take-off. The results confirm that EfficientHourglass achieved human precision in terms of annotating the body key points and joint angles. Hence, the EfficientHourglass accomplished the given hypotheses that it was able 1) to detect the ski jumper body key points and 2) identify the hip-, knee-, and ankle joint angles, both with human expert precision.

The top of the head, thorax, pelvis, right hip and left hip annotation obtained precisions ranging from 24.96% to 35.06% in HIRP as seen in Figure 10 (A). The low precision of these body key points in HIRP may be due to the ski jumper position or type of clothing, which results in a higher ME in the calculation of the joint angles. The mentioned body key points above are included in the hip joint, which was calculated using the upper body (the mean of the right shoulder, left shoulder, and upper body/thorax) and the knee annotation (the mean of both knees). The knee joint was calculated using the hip (the mean of the pelvis, right hip and left hip) and the ankle annotation (the mean of both ankles). Though the right and left shoulder obtained a close precision to the HIRP, the other body key points used in the calculation of the hip- and knee joint angles obtained a low precision, and thereby in all four models, which make them prone to a higher ME in degrees (Figure 10). The calculation of the hip joint obtained a ME with twice the magnitude compared to the ankle joint (Figure 9). The body key points used to calculate the ankle joint obtained a higher percentage in the general performance metrics and thus, obtained a lower ME in the angle calculation. Despite the low precision in the influential body key points, e.g., in the hip, the obtained ME in degrees in the joint angles predicted by the models in this study are equivalent to other studies investigating kinematic variables and joint angles. A systematic review concerning clinical assessment of gait analysis in the sagittal plane stated that estimates of data error between 2° and 5° are acceptable in a clinical setting, but may require consideration in data interpretation (39). Errors in the sagittal plane were usually around 4° (39). Another literature review of wearable technology in sport kinematics stated that an error below 5° in the sagittal plane would be of significance (15), which can be interpreted as the joint angle calculations performed by the models are within the limit of error in a clinical setting. However, previous studies on ski jumping have used a limit of error that varied between below 3° and up to 15° (5, 14). As the ankle joint angle obtained a ME of 2.4° using annotations that obtained high precision in the HIRP, it is realistic to expect the hip- and knee joint angles to obtain a similarly low ME if the precision

of the hip improves. As mentioned, this study used the mean of the occluded and visible side in the calculation of joint angles. To lower the ME in the hip-, and knee joint angle calculations explicit use of the visible side of the sagittal plane and post-processing can be used to increase precision, such as a median or low-pass filter on the marker and joint angle time series. Future studies must investigate if these methods will influence and lower the obtained ME in degrees to an acceptable limit of error for the EfficientHourglass to be implemented in analysis of ski jumping technique.

The four models underwent training on a range of image resolutions, and all four reached the inflection point around 256x256 and 288x288. Higher image resolution would result in the models using more GLOPs with only a minor improvement of performance and potentially becoming a slower motion tracker. There is a tendency of increasing precision between -B0 to -B1 and block1to5 to block1to6. A study using the same CNN architecture (28) in motion tracking of infants had a 23.5% difference in HIRP in  $PCK_h@10$ . This difference, and the models' precision, may be due to the fact that the infants' images are taken in the frontal plane, whilst the ski jumpers' images are taken in the sagittal plane. A problem with images taken in the sagittal plane is the occlusion of one side of the body. The occlusion of one side in the sagittal plane most probably affects the performance of the raters to annotate more randomly in the hope of placing the marker correctly without being able to see exactly where the body part is, though this has not been investigated. It would be interesting to see how much the visible versus occluded side affected the prediction of the body key points in terms of precision. Especially in the  $PCK_h@10$  as that is a finer precision measure used in HPE, and how that would affect the calculation of the ski jump specific angles. Though this study did not compare the precision against previous SOTA architectures, it may be similar to Groos et al. (2020) due to simpler architecture and the fact that this is also a single-person pose estimation with little to no occlusion in the analyzed image (28).

#### 4.1 Limitations of the study

Firstly, an issue with the comparison of the model precision to the HIRP is the heterogeneous number of images in the models' test subset compared to the number of images for the inter-rater spread (HIRP). The use of an explicit subset for evaluation of performance (the test subset) of the four models is considered positive (40), as it is separated from the training- and validation subset used in training of the model. However, the models' test subset includes 1864 images, while the HIRP only includes 99 images. This is a considerable difference, and in a perfect study the test subset and HIRP would have an equal number of images. Though, if each rater were to annotate 1864 images, this would be time-consuming. The precision could be affected by the "speed-accuracy trade-off" (18), i.e., if the rater would use sufficient time to ensure a precise annotation despite the increased number of images. The increased number of images could result in a lower precision, and thus, an increase in error.

Secondly, the precision in the EfficientHourglass, or other markerless motion tracking alternatives, is linked to the manual annotation of the raters due to their supervised learning. Hence, the model precision can only improve if the precision of the raters improve. A solution could be several annotations for one joint, e.g., the hip which obtained a high ME across all models (Figure 12(C)). Instead of using one annotation of the pelvis and each side of the hip, there could be two annotations in front, e.g., the spina iliaca anterior superior and greater trochanter and two in the back of the hip. Then calculate the mean of the ~four annotations and use that as one annotation. A mean of the different anatomical landmarks could ensure less inter-rater variation for the hip segment, and as

a result increase the precision of the model. Changes in annotation could also benefit for other body key points prone to low precision or high ME-h (30), such as the top of the head. As seen in Figure 10, all four models outperformed the HIRP with a lower ME-h. A possible reason for this could be the vague description of the annotation point. The depicted markers in Appendix 1 are marked in the frontal plane and could make it difficult for the raters to know which is the correct placement: mid-forehead or actual top of the head. Replacing the current guidelines with new ones seen from the sagittal plane or emphasize a more precisely written description of each body key point could be beneficial.

Thirdly, the acceptable limit of error in ski jumping has not been specified. The calculated error of in-field studies has varied between below 3° and up to 15° (5, 13, 14). One study (5) stated that the maximum difference between the in-field angles and analyzed angles did not exceed an error over 3°. The study used 2D video image data from one stationary camera placed 18 m from the edge of the jumping hill (5). Two other studies used the validity analysis of the ski jumping kinematics in take-off and early flight proposed by Chardonens et al. (14), where 75% of error of the analyzed angled were below 6° and 90% were below 15°. Due to the differences in technological appliances, there will probably be small differences between the analyzed angles (12, 14), but a narrower and more explicit limit of error must be discussed. A given limit of error is beneficial for new technology, e.g., for different SOTA CNNs, to understand how precise they must be for coaches and athletes to apply it in their analysis of technique in ski jumping. One way could be to compare the EfficientHourglass, or other markerless motion capture alternatives, against a well-established marker-based motion capture, such as Vicon or Qualisys. Some studies have validated Kinect or Organic Motion against a marker-based motion capture, and so far this has been done in gait analysis(41, 42), football (43) and as risk monitoring in sport (44). Though the results from these studies show that the markerless motion capture alternatives provide valid results in the sagittal and frontal plane, it might be challenging to apply due to the nature of ski jumping, e.g., the hill.

Fourthly, it is difficult to state which of the four EfficientHourglass models is the best performing. Larger models obtain higher precision, but also increase the use of GLOPs. The minor differences in obtained precision and ME-h, except -B0 block1to5, make it difficult to determine which to go forth with. It would be interesting to see if a re-run of the training would result in the same obtained precision in the models. Inclusion of a smaller confidence interval should be applied to see if there is a significant difference between the models to identify the optimal model. Regarding the optimal image resolution, the ankle joint was calculated using the x-coordinate (the horizontal axis) from the knee annotation. This placement differs from previous studies doing kinematic analyses, where the last annotation was placed on the back of the skis of the ski jumper (13, 45). This would only increase the size of the bounding box of the ski jumper, thereby increasing the optimal image resolution and use of GLOPs.

Fifthly, modern architectures seem to have low operational intensity (46). Further improvement of a model would be to change the architecture or choice of hardware. Using efficient hardware, such as a GPU, can speed up training and inference times (30). When designing a model architecture, consideration regarding which blocks to include and what specific task it will perform is needed. Earlier this year, Li et al. (2021) published an article on EfficientNet-X where they changed existing building blocks regularly used in CNN's with LACS (latency-aware compound scaling) and a fused convolution structure (46). LACS implement accuracy and latency as a multi-objective with compound scaling (the search for the optimal scale of depth, width and resolution), which seems to positively influence

the performance of the network (46). The article emphasizes that existing CNNs are insufficient in operational latency and low in execution efficiency. Compared to EfficientNet, the EfficientNet-X is 2x faster with similar accuracy. Compared to other architectures such as RegNet and ResNet, it is up to 7x faster (46). Whenever the pretrained blocks from the new model are released, it would be interesting to see if the architecture can improve further with an EfficientNet-X backend.

Future studies could develop a Lite-version of EfficientHourglass, as Lite-versions have shown similar precision, despite not including the same blocks or activation functions as the original network. The paper regarding EfficientNet (29) proposed the usage of EfficientNet-Lite on smaller hardware, such as a mobile device. The Lite-version included a ReLU6 activation function, instead of Swish, and removed the SE-layer. A systematic review looking at the role of wearable technology in sports stated that it is essential to make the data easy to interpret and “provide simple real-time feedback to athletes” (15). Tools easy to interpret could be especially useful for coaches or athletes to use EfficientHourglass while performing an in-hill jump.

## 4.2 Conclusion

The study validated a new type of CNN, EfficientHourglass, to annotate body key points and calculate the hip-, knee-, and ankle joint angles in ski jumping. The four EfficientHourglass models achieved human inter-rater precision (HIRP) in two of the performance metrics,  $PCK_h@30$  and  $PCK_h@error\_head$ , though this was not achieved in  $PCK_h@10$ . The hip-, knee-, and ankle joint angles obtained a ME between  $2.24^\circ$  and  $4.61^\circ$ , which is in accordance with the HIRP, and within the limits of current acceptable errors for technique in ski jumping. The model performance could be further enhanced by improvement in human annotation, exclusive use of the visual side in the sagittal plane, and post-processing of the marker and joint angle time series. Thus, with the suggested improvements, the EfficientHourglass could be utilized as a tool for athletes and coaches to analyze technique of in-hill jumps.

## References

1. Schwameder H. Biomechanics research in ski jumping, 1991–2006. *Sports Biomech.* 2008;7(1):114-36.
2. Arndt A, Bruggemann C-P, Virnavirta M, Komi P. Techniques Used by Olympic Ski Jumpers in the Transition From Takeoff to Early Flight. *J Appl Biomech.* 1995;11(2):224-37.
3. Schwameder H, Müller E. Biomechanics in ski jumping: A review. *European Journal of Sport Science.* 2001;1(1):1-16.
4. Janura M, Cabell L, Elfmark M, Vaverka F. Kinematic characteristics of the ski jump inrun: a 10-year longitudinal study. *J Appl Biomech.* 2010;26(2):196-204.
5. Janurová E, Janura M, Cabell L, Svoboda Z, Vařeka I, Elfmark M. Kinematic Chains in Ski Jumping In-run Posture. *J Hum Kinet.* 2013;39:67-72.
6. Müller E, Schwameder H. Biomechanical aspects of new techniques in alpine skiing and ski-jumping. *J Sports Sci.* 2003;21(9):679-92.
7. Virnavirta M, Isolehto J, Komi P, Schwameder H, Pigozzi F, Massazza G. Take-off analysis of the Olympic ski jumping competition (HS-106m). *J Biomech.* 2009;42(8):1095-101.
8. Ketterer J, Gollhofer A, Lauber B. Biomechanical agreement between different imitation jumps and hill jumps in ski jumping. *Scand J Med Sci Sports.* 2021;31(1):115-23.
9. Virnavirta M, Kivekäs J, Komi PV. Take-off aerodynamics in ski jumping. *J Biomech.* 2001;34(4):465-70.
10. Lorenzetti S, Ammann F, Windmüller S, Häberle R, Müller S, Gross M, et al. Conditioning exercises in ski jumping: biomechanical relationship of squat jumps, imitation jumps, and hill jumps. *Sports Biomech.* 2019;18(1):63-74.
11. Ettema G, Hooiveld J, Braaten S, Bobbert M. How do elite ski jumpers handle the dynamic conditions in imitation jumps? *J Sports Sci.* 2016;34(11):1081-7.
12. Nymoen K. *Methods and Technologies for Analysing Links Between Musical Sound and Body Motion* [Doctor thesis]. Oslo: University of Oslo; 2013.
13. Logar G, Munih M. Estimation of joint forces and moments for the in-run and take-off in ski jumping based on measurements with wearable inertial sensors. *Sensors.* 2015;15(5).
14. Chardonens J, Favre J, Cuendet F, Gremion G, Aminian K. Measurement of the dynamics in ski jumping using a wearable inertial sensor-based system. *J Sports Sci.* 2014;32(6):591-600.
15. Adesida Y, Papi E, McGregor AH. Exploring the Role of Wearable Technology in Sport Kinematics and Kinetics: A Systematic Review. *Sensors.* 2019;19(7):1597.
16. Colyer SL, Evans M, Cosker DP, Salo AIT. A Review of the Evolution of Vision-Based Motion Analysis and the Integration of Advanced Computer Vision Methods Towards Developing a Markerless System. *Sports Med Open.* 2018;4(1).
17. Cust EE, Sweeting AJ, Ball K, Robertson S. Machine and deep learning for sport-specific movement recognition: a systematic review of model development and performance. *J Sports Sci.* 2019;37(5):568-600.
18. Zimmerman ME. Speed–Accuracy Tradeoff. In: Kreutzer JS, DeLuca J, Caplan B, editors. *Encyclopedia of Clinical Neuropsychology*: Springer New York; 2011.
19. Andriluka M, Pishchulin L, Gehler P, Schiele B. 2D Human Pose Estimation: New Benchmark and State of the Art Analysis. 2014 IEEE Conference on Computer Vision and Pattern Recognition2014. p. 3686-93.
20. Lin T-Y, Maire M, Belongie S, Bourdev L, Girshick R, Hays J, et al. Microsoft COCO: Common Objects in Context2014:[arXiv:1405.0312 p.]. Available from: <https://ui.adsabs.harvard.edu/abs/2014arXiv1405.0312L>.
21. Cao Z, Hidalgo G, Simon T, Wei S-E, Sheikh Y. OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. 2018.
22. Gu Y, Zhang H, Kamijo S. Multi-Person Pose Estimation using an Orientation and Occlusion Aware Deep Learning Network. *Sensors.* 2020;20(6):1593.

23. Insafutdinov E, Pishchulin L, Andres B, Andriluka M, Schiele B. DeeperCut: A Deeper, Stronger, and Faster Multi-Person Pose Estimation Model 2016. Available from: <https://ui.adsabs.harvard.edu/abs/2016arXiv160503170I>.
24. Newell A, Yang K, Deng J. Stacked Hourglass Networks for Human Pose Estimation 2016: [arXiv:1603.06937 p.]. Available from: <https://ui.adsabs.harvard.edu/abs/2016arXiv160306937N>.
25. Giles B, Kovalchik S, Reid M. A machine learning approach for automatic detection and classification of changes of direction from player tracking data in professional tennis. *J Sports Sci.* 2020;38(1):106-13.
26. Chen L, Wang W. Analysis of technical features in basketball video based on deep learning algorithm. *Signal Processing: Image Communication.* 2020;83.
27. Hendry D, Chai K, Campbell A, Hopper L, O'Sullivan P, Straker L. Development of a Human Activity Recognition System for Ballet Tasks. *Sports Med Open.* 2020;6(1):10.
28. Groos D, Adde L, Støen R, Ramampiaro H, Ihlen EAF. Towards human performance on automatic motion tracking of infant spontaneous movements 2020 October 01, 2020: [arXiv:2010.05949 p.]. Available from: <https://ui.adsabs.harvard.edu/abs/2020arXiv201005949G>.
29. Tan M, Le QV. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks 2019 May 01, 2019: [arXiv:1905.11946 p.]. Available from: <https://ui.adsabs.harvard.edu/abs/2019arXiv190511946T>.
30. Mathis A, Schneider S, Lauer J, Mathis MW. A Primer on Motion Capture with Deep Learning: Principles, Pitfalls, and Perspectives. *Neuron.* 2020;108(1):44-65.
31. Chollet F. *Deep Learning with Python: Manning Publications* 2017.
32. Yang W, Li S, Ouyang W, Li H, Wang X, editors. *Learning Feature Pyramids for Human Pose Estimation.* 2017 IEEE International Conference on Computer Vision (ICCV); 2017 22-29 Oct. 2017.
33. Hu J, Shen L, Albanie S, Sun G, Wu E. Squeeze-and-Excitation Networks 2017. Available from: <https://ui.adsabs.harvard.edu/abs/2017arXiv170901507H>.
34. Ioffe S, Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift 2015: [arXiv:1502.03167 p.]. Available from: <https://ui.adsabs.harvard.edu/abs/2015arXiv150203167I>.
35. Ramachandran P, Zoph B, Le QV. Searching for Activation Functions 2017 October 01, 2017: [arXiv:1710.05941 p.]. Available from: <https://ui.adsabs.harvard.edu/abs/2017arXiv171005941R>.
36. Deng J, Dong W, Socher R, Li L, Kai L, Li F-F. ImageNet: A large-scale hierarchical image database. 2009. p. 248-55.
37. Kingma DP, Ba J. Adam: A Method for Stochastic Optimization 2014 December 01, 2014: [arXiv:1412.6980 p.]. Available from: <https://ui.adsabs.harvard.edu/abs/2014arXiv1412.6980K>.
38. Chen Y, Tian Y, He M. Monocular human pose estimation: A survey of deep learning-based methods. *Computer Vision and Image Understanding.* 2020;192:102897.
39. McGinley JL, Baker R, Wolfe R, Morris ME. The reliability of three-dimensional kinematic gait measurements: A systematic review. *Gait Posture.* 2009;29(3):360-9.
40. Halilaj E, Rajagopal A, Fiterau M, Hicks JL, Hastie TJ, Delp SL. Machine learning in human movement biomechanics: Best practices, common pitfalls, and new opportunities. *J Biomech.* 2018;81.
41. Perrott MA, Pizzari T, Cook J, McClelland JA. Comparison of lower limb and trunk kinematics between markerless and marker-based motion capture systems. *Gait Posture.* 2017;52:57-61.
42. Kanko RM, Laende E, Selbie WS, Deluzio KJ. Inter-session repeatability of markerless motion capture gait kinematics. *J Biomech.* 2021;121.
43. Kotsifaki A, Whiteley R, Hansen C. Dual Kinect v2 system can capture lower limb kinematics reasonably well in a clinical setting: concurrent validity of a dual camera markerless motion capture system in professional football players. *BMJ Open Sport Exerc Med.* 2018;4(1).

44. Johnson WR, Mian A, Lloyd DG, Alderson JA. On-field player workload exposure and knee injury risk monitoring via deep learning. *J Biomech.* 2019;93:185-93.
45. Bessone V, Petrat J, Schwirtz A. Ski Position during the Flight and Landing Preparation Phases in Ski Jumping Detected with Inertial Sensors. *Sensors* 2019;19(11).
46. Li S, Tan M, Pang R, Li A, Cheng L, Le Q, et al. Searching for Fast Model Families on Datacenter Accelerators 2021 February 01, 2021:[arXiv:2102.05610 p.]. Available from: <https://ui.adsabs.harvard.edu/abs/2021arXiv210205610L>.



## Appendices

Appendix 1: Guidelines for human annotation

Appendix 2: Further explanation of the different layers and functions in CNN and EfficientHourglass


# Appendix 1: Guidelines for human annotation

6.9.2019




annotation/README\_ski.md at master · daniegr/annotation

Branch: master ▾ Find file Copy path

annotation / README\_ski.md

 daniegr Merge branch 'master' into expand  
8a4e25c 1 minute ago

1 contributor

Raw Blame History   

253 lines (88 sloc) 6.06 KB

## Annotation Program

*Tool for annotating video frames*

### Using Annotation Program

#### How to use the annotation program?

1. Create shortcut of program

If this is not already done, as a first step you should create a shortcut or alias of the program to make it easily accessible.

- macOS: Navigate to the `source` folder from the `program` folder and locate the file called `annotate`. Make an alias of this file and place it inside the `program` folder. Do the same for the files called `annotations_ski.csv` and `sessions_ski.txt`.
- Windows: Navigate to the `source` folder from the `program` folder and locate the file called `annotate.exe`. Make a shortcut of this file and place it inside the `program` folder. Do the same for the files called `annotations_ski.csv` and `sessions_ski.txt`.

2. Run program

If you have successfully created a shortcut or alias, the program can be run in the following way.

- macOS: Run `annotate` in `program` folder
- Windows: Run `annotate.exe` in `program` folder

3. Annotate video frames

In order to guide the user of how the annotation task should be ideally performed, there are some training examples to begin with. The training examples is completed before the real annotation work can begin. For each training example the following steps must be performed:

- i. Click on the next body part according to the help text and the guideline in the bottom left corner.
- ii. A marker with a grey color indicates that the marker is not correctly placed. Drag the marker to adjust its location.
- iii. When the correct location is assigned to a marker, the marker is displayed in the color corresponding with the guideline. Accordingly, the next body part can be placed by repeating the procedure.
- iv. When all body part markers have been correctly placed, **CONFIRM ANNOTATION** should be pressed and the next training example will pop up
- v. After completing all training examples the user is displayed a message and should press the button **START TO ANNOTATE** to start annotating.

After completing training, the user starts annotation randomly displayed video frames:

[https://github.com/daniegr/annotation/blob/master/README\\_ski.md](https://github.com/daniegr/annotation/blob/master/README_ski.md)

1/4

- i. Similarly to during training, the body parts are clicked in a specific sequence indicated by the help text and the guideline of the bottom left corner of the annotation GUI.
- ii. At any time body parts can be clicked and dragged to adjust the placement of markers.
- iii. When all markers have been placed and the user is satisfied with the placement of markers the annotation is saved by pressing **CONFIRM ANNOTATION**. The next frame is then displayed to the user and the process of annotating a frame is repeated.
- iv. If the user at any time wants to change a previously annotated frame the button **LAST FRAME** can be pressed and adjustments made accordingly. The updated locations of markers will replace the ones that already exist.

A more thorough description of the functionality of the annotation GUI as well as of how to annotate correctly is given below.

#### 4. Close program

The program can be closed at any time by pressing the **ESCAPE** key and restored by running the program as described above.

#### 5. Backup annotations

The file `annotations_ski.csv` contains the coordinates of the annotated frames. In order to ensure that this valuable data is not lost in case of losing the hard drive containing the program, establish routines for copying the `annotations_ski.csv` file to your computer or a personal cloud location.

#### Definition of body parts



#### General comment

All body parts should be annotated in all frames. If a body part overlaps another or the body part is not visible in the frame, the marker should be placed where the user believes the body part to reside in the 2D-plane. Although this can result in markers being placed on top of each other, this is consistent with the guidelines of annotation.

The final coordinates is recorded from the center of the body part markers. Hence it is important to be precise when placing the markers.

#### 1. Head top

- Top of the forehead

#### 2. Upper neck

- Center of the larynx (adamseple)

**3. Right shoulder**

- Center of the right shoulder joint

**4. Right elbow**

- Center of the right elbow joint

**5. Right wrist**

- Center of the right wrist joint

**6. Upper chest**

- Midway between the center of the left and right shoulder

**7. Left shoulder**

- Center of the left shoulder joint

**8. Left elbow**

- Center of the left elbow joint

**9. Left wrist**

- Center of the left wrist joint

**10. Mid pelvis**

- Midway between the left and right pelvis (see definition of left and right pelvis below)

**11. Right pelvis**

- Right spina iliaca anterior superior (fremre øvre bekkenkant)

**12. Right knee**

- Center of the right knee joint

**13. Right ankle**

- Center of the right ankle joint

**14. Left pelvis**

- Left spina iliaca anterior superior (fremre øvre bekkenkant)

**15. Left knee**

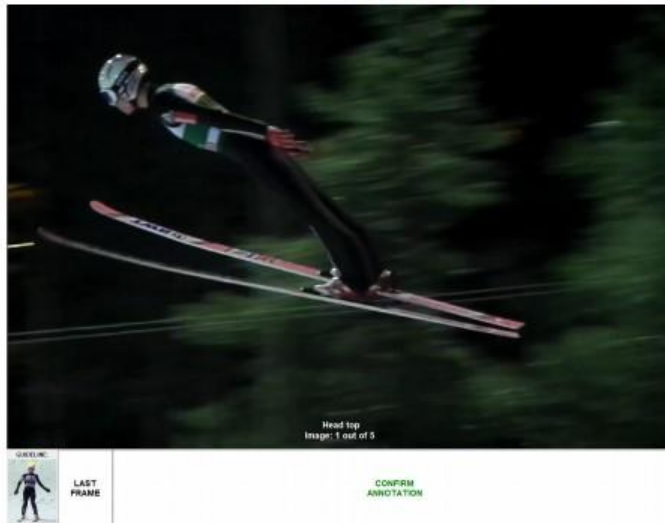
- Center of the left knee joint

**16. Left ankle**

- Center of the left ankle joint

**Program functionality**

GUI



- **GUIDELINE:** displays the order of how the body parts should be clicked and which colors that belong to which body parts

#### Functions

##### Button actions

- **LAST FRAME:** button to move to the last frame that was annotated
- **CONFIRM ANNOTATION:** button to confirm annotation and move to the next frame

##### Mouse actions

- **CLICK** on frame: Place marker on the next body part to annotate
- **DRAG** existing marker: Adjust location of existing marker
- **RIGHT CLICK** on frame: Same as **CONFIRM ANNOTATION**

##### Keyboard actions

- **RIGHT ARROW:** Same as **CONFIRM ANNOTATION**
- **ENTER:** Same as **CONFIRM ANNOTATION**
- **SPACE BAR:** Same as **CONFIRM ANNOTATION**
- **LEFT ARROW:** Same as **LAST FRAME**
- **BACKSPACE:** Same as **LAST FRAME**
- **ESCAPE:** Close the program

## Appendix 2: Further explanation of the different layers and functions in CNN and the EfficientHourglass

### A

Activation functions (Swish): Other layers in a CNN are linear functions. For the model to learn complex features, a non-linear function is needed. This is where the activation function comes in. The negative input is transformed to a value close to zero, whilst positive input is unchanged. The activation function helps to decide if a neuron would fire or not (35).

Adam optimizer: Adam stands for Adaptive Moment estimation. The optimizer wants to minimize the loss function (37). Appliance of momentum to not get stuck in the local minimum and find the global minimum of the loss function. See figure 1.

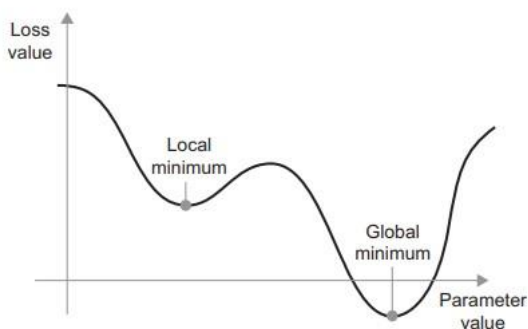


FIGURE 13: ILLUSTRATION OF THE LOSS FUNCTION AND OPTIMIZER, FROM CHOLLET (2017) (31).

Add-layer: to keep salient features "in the loop". By applying the add-layer, the earlier feature maps can be further used in the training of detecting the body key points.

Attention (and gating) mechanism: a mechanism used in the SE-layer. Scaling the more important channels to a higher value. Attention to the salient features relevant for the task (33). Suppressing feature activations in irrelevant regions.

Average pooling: applied after a convolutional layer. Used to reduce the spatial size of the convolved feature map, depth remains the same. By reducing size of feature map → decrease computational power required to process the data. Returns the average of all values from the portion of the feature map (30).

### B

Backpropagation (BP) algorithm: after each forward-pass of all iterations in the dataset, the BP algorithm use the final loss value and work backwards from the top to the bottom layers (31). Applying the chain rule (gradient) + learning rate to compute the contribution that each parameter had in the loss value. New weights → (old weights) + (learning rate) \* (gradient).

Batch normalization (BN): standardizes the input to a layer for each mini-batch. Allows each layer of a network to learn by itself a little bit more independently of other layers. In each mini-batch the values are normalized in respect of the batch inputs. The BN calculates the mean and the standard deviation of the batch "at hand". Doing so speeds

up the training, forms a smoother optimization landscape and decreases the importance of the initial weights (31).

Bottom-up approach: used in multi-person pose estimation. First identifies all the body key points in the image, then group them into person specific key points (21).

Bridge blocks: Found in block 2, 3 and 5 depending on the model. Takes out detailed information for later use. A connection between encoder part and the decoder part (transpose convolution), as the image resolution goes from low-to-high. For the network to access feature maps in blocks in the encoder part.

## C

Confidence maps: the output, the annotated body key points on the ski jumper, performed by the models. Use feature maps from the bridge blocks and add-function to predict body key points correctly.

Convolution: a mathematical operation. E.g., a 128x128 input image multiplied with a 3x3 filter. Elementwise multiplication over the input image. This multiplication results in one pixel of the feature map (31).

Convolutional neural network: differ from other machine learning approaches as the CNN learns data incrementally layer by layer, and these incremental representations are learned jointly. The architecture of CNNs is inspired the human brain and its structure. Learned patterns from a convolution layer are translation invariant. After learning, the pattern can be recognized anywhere in the image (31). Successive layers of representations, and structured in literal layers stacked on top of each other (31). The main purpose is to downscale (encoder) the input image to a form which is easier to process and retain features that can be used in the upscaling (decoder) which are critical for a good prediction of the body key points.

Convolutional layer (convL): a type of filter, multiplies a set of weights with the given input image. Results in a feature map. Automatically learn features (31). Can either increase the depth of the receptive field or decrease the depth of the model depending on the size of the convolutional layer applied. 1x1, 3x3 and 5x5.

## D

Data augmentation: to prevent an overtrained model, data augmentation is performed on the training subset. This could be rotation, flipping or scaling of the given input image during training (30).

Decoder part: upsample the feature representation in the network to a desired image resolution, the output image. Low-to-high image resolution. Also known as transpose convolution (30).

Derivative: a continuous, smooth function. A small change in  $x$ , results in a small change in  $y$ .

## E

Epoch: one epoch is when the whole training sub-set has been passed forward and backward through the CNN algorithm once. A training subset goes through several epochs, this makes it possible for the CNN to readjust the weights (31).

Encoder part: downscale the input image resolution in the network. Goes from high-to-low image resolution. More computer efficient to look at a small part of an image, compared to the input image resolution. The encoder part can be seen as a magnifying glass moving over the image (30).

## F

Feature: an individual measurable property of the data (31). A feature could be colour, edge or size.

Feature map: the result of a filter. The feature map accentuates the unique features from the original image and assigns what is important in the image to classify the body key points correctly (31).

Filter: feature extraction → different filters extract different features. A set of weights, moves across the image, and systematically applied to the input image and results in a respective feature map (31). Detects spatial hierarchies of patterns in the input image.

Fine-tuning: using a pretrained model (i.e. MPII or ImageNet) can improve computer efficiency of the model as well as improve the overall precision (30). Earlier layers in the CNN detect more generic, reusable features, while deeper layers detect more specialized features. By slightly adjusting the more abstract features of the model being reused to make it more relevant for the task at hand.

FLOPs: floating point operations per second. A measure of the computer performance. FLOPs often on the x-axis plotted against a performance metrics, such as ME-h or PCKh@ $\tau$  on the y-axis.

## G

Gradient: adaptive learning rate. If the gradient is positive, it indicates an increase in weights. If the gradient is negative, it indicates a decrease in weights. The partial derivatives are collectively called the gradient (31). As it is iterative, it needs to get results multiple times to become optimal. The gradient use information from the learning rate and loss function.

Global minimum: a term used when talking about the loss function. An optimally trained model has obtained a global minimum of the loss function in the validation subset. See illustration under "ADAM optimizer".

GLOPs: giga (billion) floating points per second.

## H

Hourglass architecture: consists of an encoder and decoder part, which is a high-to-low resolution network in the encoder part. Low-to-high resolution network in the decoder part. Combines features captured across different image resolutions. Including bridging convolutional layers between the encoder and decoder part (24)

Human pose estimation (HPE): a complex task of detecting and connecting body key points on a human body to understand their pose (24). Localization of human joints in an image or video sequence.



## I

ImageNet: a large database containing images of humans in different activities and situations. The database is organized in a hierarchical WordNet, meaning the images are described in words and sentences. These are called synsets, and each synset in ImageNet has at least 1000 pictures. The images are quality-controlled and human-annotated (36).

Inter-rater error: a collected mean error for the manual annotation performed by human raters. Calculated to see how similar the human raters annotate compared to each other. Seen as the "gold standard" for the network. All predictions made by the network is compared to the manual annotation.

Iterations: the ski jumper dataset of 9324 images were split into mini-batches of 16 images in each.  $9324/16 = 419$  iterations. Each iteration repeats the similar process to achieve a desired goal (minimum loss function) where the result of one iteration is the starting point for the next (31).

## L

Loss function: computes a distance score of the prediction of the network and the ground truth (the human inter-rater precision). The loss function captures how well the network has done on the specific task. The score is used as a feedback signal used for learning and represents a measure of success for the task at hand (31).

Learning rate: a set value between 0 and 1 multiplied with the specific value of a gradient. Can be seen as the magnitude of change of the model weights after each iteration to achieve a minimum of the loss function during training (30). In EfficientHourglass, the learning rate was set to 0.001.

## M

Mini-batch: in a big dataset the whole dataset/batch is divided into several smaller mini-batches. In EfficientHourglass, the mini-batches are set to 16 images in each.

Mobile inverted bottleneck sub-blocks: called MBConv1 and MBConv6. Expand the number of channels  $\rightarrow$  image is height (H) \* width (W), \* channels (C) (28). The stages are described below:

- 1) 1x1 conv inverted bottleneck
- 2) Depth-wise convolution
- 3) SE-layer
- 4) 1x1 conv bottleneck

MPII Human Pose Estimation algorithm: a dataset trained on the ImageNet database. For evaluation of articulated human pose estimation. It consists of  $\sim 25\ 000$  images of humans in different human activities seen from different angles and light (19). In EfficientHourglass, the pretrained blocks from MPII are used in the encoder part.

## N

Neuron: Learnable weights. Computes a dot product of the weights after an applied filter (31). Connected to the receptive field.

## O

Optimizer: in EfficientHourglass, the ADAM optimizer is implemented. The optimizer is a type of algorithm used to update the network weights and learning rate in order to reduce the loss function (31).

## R

Rhb\_index: The filenames without information on hill or person were separated from the phb\_index. Each random combination was given a unique number, e.g., for NM20PL\_Mbib87 and SJs2. The percentage of images in each subset was calculated after phb\_index, and the needed amount to achieve the desired percentage was supplemented from the rhb\_index sheet.

Receptive field: is a patch of the whole input image, i.e., the information a neuron has access to. A large enough receptive field is important for the network to detect features to predict precisely. To increase the receptive field: more conv. layers (deepen the network), depth-wise convolution and use of bridge blocks (different receptive fields across different ranges of image resolutions) (30).

## P

Parameters: also known as weights.

PCK: percentage of correct key points. Defined as the fraction of predictions residing within a given distance, in this study the size of the head. "If a predicted joint falls within a threshold of the ground-truth joint location, it is counted as a true positive" (38).

## S

Swish: the activation function used in the EfficientHourglass. A non-linear function. Not only positive values, but also small negative values for the non-monotonicity. Unbounded above, bounded below (35).

Stride: the filter step size. Moves across the input image.

Squeeze-and-excitation layer: assign weight to feature maps based on their relevance for a given output. Puts attention to more relevant features and suppress less useful ones (33).

1. Input image is a tensor with (height(h) + weights(w)) + channels(c)
2. Global average: multiply h\*w to get 1 pixel → average of feature maps
3. 1 x 1 x c is the "new" tensor
4. A 1x1 convL is performed on the 1 x 1 x c
5. The channel weights are assigned a value between 0 and 1, with help from the Swish activation function
6. The old tensor is multiplied to the new values
7. Attention → more efficient for the model training → what is interesting for the model to focus on

## T

Transfer learning: a model trained for a task is reused as the starting point for another model on a second task. Such as using the MPII as benchmark in EfficientHourglass,

which is pretrained on ImageNet. Increases computer efficiency, reduces time resources and can increase the precision of the retrained model.

Transpose convolutions: the same as the decoder part. Upscale the image and going from low-to-high resolution. Uses feature maps from the encoder part.

