End-to-End Pose Estimation from Monocular Ice-hockey videos

Jerrin Bright*
Systems Design Engineering
University of Waterloo
jerrin.bright@uwaterloo.ca

Bavesh Balaji*
Systems Design Engineering
University of Waterloo
bbalaji@uwaterloo.ca

Harish Prakash*
Systems Design Engineering
University of Waterloo
h3prakas@uwaterloo.ca

***Abstract*—Estimating human pose keypoints from a monocular video is a challenging task, especially in agile environments. Most of the previous research works use high-resolution inputs with ample 3D groundtruth to train their model. In this work, we propose a unique solution that can efficiently estimate the 3D pose of ice-hockey players, which ultimately helps in action recognition and analytics of the players without 3D pose groundtruth data. We further propose a 'teacher-student' modeling of 3D networks, to counter the ill-poised problem of 3D ground truth data scarcity. Feeding in our fine-tuned model's 2D predictions, we acquire 3D predictions from the 'Teacher' network, which serves as the groundtruth for our 'Student' network, forming a feedback loop of refinement and predictions. In this work, a novel student estimator network is proposed to induce inductive bias into transformers. Experimentation with various 2D and 3D pose estimators has been done. A modular architecture has been proposed that gives robust results irrespective of occlusions and depth ambiguities. The qualitative results of the implemented 2D and 3D pose estimators can be viewed from https://pose-estimation-videos/.**

***Index Terms*—Pose Estimation, Sports Analysis, Convolutional Neural Networks, Transformers**

I. Introduction

Human action recognition has been one of the most researched problems in computer vision. The most successful approaches model this problem as a function of pose estimation, based on the keypoint representation of human joints. Pose Estimation, in general, refers to predicting the pose of an entity (humans, in most cases), and helps to determine their physical orientation with respect to an environment. This plays a vital role in sports analytics, in estimating the 'correct pose' of sportsmen to decode their style of play, analyze their actions, avoid injuries, plot their activities, and for a gamut of other use cases. Vision-based predictions of poses are especially relevant, due to their minimally invasive approach and methodology.

But, there exist certain constraints in vision settings, which include occlusion, foreshortening, shadows, depth ambiguity, and misdirection which results in inconsistent pose estimates, especially in uncontrolled agile environments. This is easily observed in fast-paced team sports such as Ice-hockey, basketball, soccer, rugby, etc., where there exist several constraints to estimate the relevant pose of a player. This has proven to be an ill-poised problem in vision.

* These authors contributed equally towards this course project

In our work, we study the different approaches to mitigate this, by an end-to-end implementation and comparative study of novel state-of-the-art architectures for 2D Pose Estimation. To this end, we utilize a novel Ice-Hockey Dataset, which contains manually annotated 2D ground truth keypoints and bounding boxes, sampled at 30 frames per second. We opt for the top-down approach (human bounding-box detection followed by pose estimation) as it has proven to be the most efficient and handpick four models [1]–[4] based on our use-case.

In the 3D space, existing approaches for specific use cases fine-tune a pre-trained model, if target 3D keypoints are available. But, for our dataset, there were no 3D ground truth targets available during the time of this work. To overcome this constraint, we formulate a unique approach using "Teacher-Student" networks, whereby, a state-of-the-art network acts as the Teacher, whose predictions are fed as the groundtruth to the relatively under-performing Student network, creating a feedback loop of learning. Our novelty lies in designing a 'Transformer-Convolution' (Trans-Conv) embedded architecture for the Student network, which models 3D keypoints based on both transformer-based and convolution-based architectures.

Our approach also overcomes another bottleneck in vision, which is the unavailability of high-fidelity ground truth annotations for supervision. This is especially predominant in manual annotations of monocular broadcast videos, due to motion blurring and constant occlusions. Our architecture factors in the temporal domain (receptive field > 1) [5], mitigating both these constraints, by extrapolating the poses across the sampled frames to account for occlusion and blurring. To summarize our contributions:

- We pursue a comparative study of top-down 2D Pose estimation architectures,
- We model the 3D space as a 'Student-Teacher' relation,
- We propose a novel 3D Pose estimation architecture (Trans-Conv), and
- We utilize a novel Ice-hockey dataset with annotated bounding boxes and keypoint coordinates.

The rest of the paper is structured as follows. In Section II, previous research works pertaining to pose estimation and sports analytics are discussed. In Section III and IV, the

adapted 2D and 3D pose estimation techniques are explained in detail. In Section V, the proposed architecture is explained. The dataset and metrics used in this work are briefed in Section VI and VII. Then, the experimentation done for 2D and 3D pose estimation techniques is explained in Section VIII. Finally, the work is concluded in Section IX along with the future works in Section X.

II. Related Works

Human Pose estimation is a fundamental problem in computer vision that has been researched for a long time. Classical approaches to pose estimation include pictorial structures models [6] and Flexible mixture-of-parts [7]. These frameworks broadly use tree-based probabilistic graphical approaches to model the spatial relationships between different joints. Other classical approaches involve extracting important features through various feature extraction techniques such as contour detection, color histograms, and histogram-of-gradients (HOG) [8]. However, these approaches were not able to handle occlusion and model the spatial information effectively.

The advent of deep learning and convolutional neural networks helped in efficient feature encoding and better generalization. Hence, a multitude of works has been conducted on the use of deep learning approaches for pose estimation. Toshev et al. [9] initially formulated pose estimation as a regression problem of finding body joints, and used CNNs to estimate the poses. Pischulin et al. [10] followed a bottom-up approach by detecting all the keypoints first using CNNs, and then using ILP to cluster the keypoints. Newell et al. [11] was the first to use a multi-stage architecture where each stage consisted of repeated down and upsampling layers with skip connections [12] to extract as much information as possible. Subsequently, a lot of architectures [1], [2], [13], [14] use the multi-stage technique and follow a top-down approach.

Recently, transformer architectures have gained a lot of traction in various computer vision tasks. Most models [4], [15], [16] incorporate CNN backbones to extract features and then employ transformer encoder and/or decoder layers to refine the features. On the other hand, HRFormer [17] and ViTPose [18] directly use transformers to extract features and predict keypoints.

In the 3D pose estimation space, several attempts have been made to extrapolate 2D keypoint coordinates into 3D, either through multi-view geometry [19] or Temporal Convolutions [5]. [19] uses 2D poses from multi-view images to obtain 3D poses and camera geometry using Epipolar geometry. [5] use an end-to-end Temporal-Convolutional Neural architecture to predict 3D poses from videos, using self-supervised training. Yujun et al. [20] utilize a Graph-based Convolutional Network, to explicitly incorporate human body dimensions into 3D predictions for plausible results. [21], following a similar graph-based approach, predicts 3D keypoints based on the correlation between each individual keypoint with all others, to model occlusion.

Recently, Transformer-based approaches to vision have become predominant since their widespread success in Natural Language Processing tasks, like BERT (Bi-directional Encoder Representations from Transformers) [22], GPT (Generative Pre-trained Transformer) [23], and Dall-E [24]. A range of vision tasks including Classification [25], Detection [26], Segmentation [27], Tracking [28]–[30], and Pose Estimation [4], [31], [32] have achieved SOTA results with Transformer-based architectures. This motivated us to also leverage a self-attention-based neural architecture rather than just Convolutions for 3D Pose Estimation.

A. Pose estimation for Ice-hockey

Multi-stage architectures following the top-down approach have shown great potential in ice-hockey. Fani et al. use a stacked hourglass network to predict the 2D keypoints [33] and subsequently integrate it with a feature transformer to perform action recognition [34]. Furthermore, Neher et al. [35] use 2 stacked hourglass networks, one pretrained on 16 joints and another untrained to find out the poses of hockey players along with their sticks (18 joints). More recently, McNally et al. [36] incorporates neural architecture search to design efficient pose estimation and accelerate the search using a novel weight transfer method.

III. 2D Pose Estimation

A. MultiStage Pose Network

Multi-Stage Pose Networks [1] adopt the top-down approach in two steps. In the first step, manual annotations of all the players on the rink are used to crop the input frames and create multiple images consisting of a single person. The pose estimation network then uses repeated down and up-sampling to continuously refine the estimation of poses. This network mainly proposes three major design improvements on other multi-stage networks.

The first one is the equal channel width design followed in all the networks. All the existing networks use the same number of channels in each level of a downsampling module. However, this reduces the size of the feature map as we go down a single stage, making it more difficult to capture relevant information. To solve this problem, this network doubles the number of channels(convolutional kernels) at every level of a downsampling module, maintaining the size of the feature map throughout the stage. This helps the model capture more information in the downsampling module, resulting in better localization of keypoints.

The second improvement made by this architecture is the cross-stage feature aggregation. This network enables us to propagate the features extracted during the initial stages by aggregating them with the features in successive stages. This helps in retaining a lot of information without adding a lot of layers, making the model more robust and foolproof.

The third and most important improvement made by the model is the coarse-to-fine supervision. At the end of every stage, the outputs are converted into gaussian heatmaps and compared with ground truth heatmaps to refine the localization accuracy. In this network, they perform this intermediate supervision at every stage using decreasing gaussian kernel

sizes (instead of using same size kernels at every stage) as this gives a more accurate estimate of the features extracted.

B. High Resolution Network

High Resolution Network [2] is a multi-stage pose estimation network that focuses on producing and maintaining accurate high-resolution relationships. This network starts by extracting features at a higher resolution and then goes on reducing the resolution as we go deeper into the architecture. The one unique aspect of this network is the parallel connections across different resolutions, in comparison to the serial connections that are used in other pose estimation architectures. These parallel connections help in effectively maintaining the high-resolution features extracted at the start of the network without losing important information. The other differentiating factor of this model from other existing models is the repeated multi-scale fusion across different resolutions of a single stage. This novel technique aggregates features from different resolutions by either upsampling(using nearest neighbor interpolation followed by 1x1 convolutions) or downsampling(using 3x3 strided convolutions). This concatenation of features within a single stage helps in producing refined and robust representations.

C. Distribution-Aware coordinate Representation of Keypoint

The state-of-the-art pose estimation models do not directly take the 2D coordinates of each joint as their input and predict 2D coordinates for every keypoint in a given image. This is mainly because the 2D coordinates do not contain any spatial and contextual information, making pose estimation extremely challenging. Hence, all the existing networks convert these 2D coordinates to heatmaps using a gaussian kernel to gain some much-needed spatial information. This work [3] focuses on improving this encoding and subsequent coordinate decoding part(after the predictions made by the network) and provides a more principled distribution-aware method.

The standard coordinate encoding methods downsample bounding boxes to a smaller dimension, transforming the ground-truth coordinates accordingly. This downsampling is defined as shown in equation 1.

$$g' = (u', v') = g/\lambda = (u/\lambda, v/\lambda) \tag{1}$$

In equation 1, $g = (u, v)$ as the ground-truth coordinate and λ is the downsampling ratio. After performing this downsampling, standard methods generally quantize these coordinates using the floor or the ceil function to facilitate kernel generation. However, this causes an inaccurate and biased representation because of quantization error. This work solves that problem by eliminating the quantization step and placing the center of the heatmap at the downsampled coordinate g'.

The standard coordinate decoding methods find the coordinates of the maximal and second maximal activation. The joint location is then predicted by shifting the location of the maximal activation 0.25 pixels towards the second maximal activation. This shifting is done to compensate for the quantization error. However, this is an empirical method that is found to have success but does not have any intuition behind it. Also, the predicted heatmap is not exactly gaussian in nature and hence, the point with the maximal activation may not be estimated accurately. Hence, this work proposes a theoretically sound method to explore the entire heatmap and find the underlying maximal activation by modulating the heatmap to make it gaussian and then using that fact along with the Taylor series to find the actual coordinates of the keypoint.

D. TransPose

This network [4] leverages the recent success of transformers in computer vision tasks and replicates it in pose estimation. More specifically, this network uses a common CNN backbone such as ResNet or HRNet to extract the features from the input image. The extracted features are then passed through N transformer encoder layers where each layer consists of a multi-self-attention head, layer normalization, and feed-forward neural networks. The attention layers help in capturing long-range relationships and reveal the dependencies that determine the location of the maximal activation. The N different attention layers capture different positions of maximal activation corresponding to different joints. The final attention layer acts as an aggregator, which collects contributions from image clues and forms the maximum positions of keypoints. The output from the final encoder layer is then passed through a regression head to find the heatmap for every keypoint.

IV. 3D Pose Estimation

A. Multi-Hypothesis Former

Li et al. [32] proposed MHFormer which leverages vision transformer networks and the key idea is to learn spatio-temporal representation of pose hypotheses and is done using a three-stage process. It starts by creating multiple initial representations, then self-communicating which is then followed by cross-communicating between the hypotheses to get accurate predictions.

In the first stage, Multi-Hypotheses Generation (MHG) is done where the intrinsic structure information of the human joints is modeled and several multi-level features are generated in the spatial domain. Every single-hypothesis feature is refined by the second stage, the Self-Hypotheses Refinement (SHR) stage. The SHR consists of two blocks- Multi-Hypotheses Self Attention (MHSA) and Multi-Layer Perceptron (MLP). In MHSA, feature enhancement is accomplished, by passing messages within each hypothesis, and the MLP exchanges information across the generated hypothesis from MHG. These different hypotheses are then merged to get a converged hypothesis which is then diverged again to generate multiple hypotheses.

The third stage, Cross-Hypotheses Interaction (CHI) is introduced here to enhance the accuracy by leveraging the information between different generated hypotheses. It uses Multi-Hypothesis Cross Attention (MHCA) that cross-communicates

with various hypotheses for better interaction modeling. Following MHCA, an MLP is used just like after MHSA, to merge all the diverse hypotheses. The high performance of the MHFormer network comes with the high computational cost of the network.

B. Strided Transformer Encoder

Li et al. [31] proposed STE which leverages a sequence of 2D pose data as input and outputs the target 3D pose. It uses two transformer encoders to capture the global and local contexts of the input 2D joint data. Firstly, position embedding on the input 2D poses is done and is fed to the Vanilla Transformer Encoder (VTE) where Multi-Head Self-Attention and a Convolutional Feed Forward Network (CFFN) are used to capture the long-range dependencies of the 2D pose sequences.

The output of the VTE is passed as input to the Strided Transformer Encoder (STE) which uses the same MHSA as the VTE but has tweaked the vanilla CFFN. The feed-forward network used in STE has 1D convolution in place of the fully connected layers with a striding factor of S. This tweak in the STE helps to merge the sequences of the nearby poses and improves capturing the local context of the 2D poses. The output of the STE block is then connected to the regression head which gives the reconstructed target 3D pose.

C. Graph Attention Spatio-Temporal Network

Liu et al. [21] proposed a graph-based architecture to learn kinematic connections, constraints, and symmetry of humans by capturing the temporal and spatial information using attention mechanisms. The architecture consists of three major modules- temporal convolutional layers, the local spatial attention graph, and the global spatial attention graph. Considering the input 2D pose as a graph, the pose joints are represented as the nodes, and the lines that connect the skeletons are the edges of the graph.

Given a 2D sequence of poses, firstly, convolution is done and is sent to the graph attention block, which consists of the local and global spatial attention graph. This is followed by temporal convolution. This cycle is then done multiple times until the final 3D pose is obtained from the sequence of 2D poses. The temporal convolution consists of convolutions and dilation which ultimately increases the receptive field without increasing the number of weight parameters by inducing gaps in the kernel.

The local spatial attention graph captures the correlation between each and every vertice (the 2D joint pose) with its corresponding vertices. The global spatial attention graph captures the correlation of every vertice with every other vertice of the human 2D pose. This helps in capturing the global pose context and makes the pose estimation robust to occlusions and depth ambiguity.

V. Our Methodology

In this section, the modular components of the proposed human pose estimation technique are explained meticulously. The overview of the proposed network is shown in Figure 1.

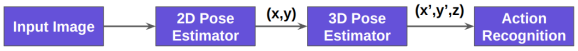


Fig. 1. Overview of the Adapted Technique

First, to estimate the 2D pose of the players for our custom dataset, we optimized the four state-of-the-art (SOTA) 2D pose estimation networks explained in Section III. We fine-tuned these SOTA models for our ice-hockey dataset and compared the performance and robustness of these models.

Then the obtained 2D pose keypoints of each player are fed as input to the 3D pose estimator module. The 3D pose estimator module is shown in Figure 2.

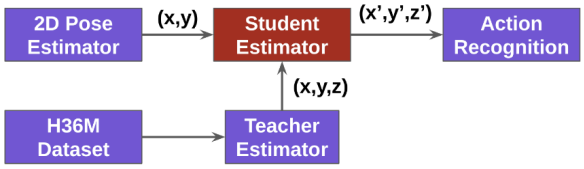


Fig. 2. Overview of the Adapted Technique

In 3D pose estimation, the unavailability of 3D Ground Truth Keypoints to supervise training is an ill-poised problem. Since our dataset falls under this category and doesn't have 3D annotations, we had to train our network without 3D ground truth pose keypoints. Thus, we came up with a teacher-student estimator technique where the input frames (from the ice-hockey dataset) were inferred on the teacher estimator network which was trained on the Human3.6M dataset [37]. These inferred 3D pose keypoints were fed as input to the student estimator network along with the 2D poses generated using our dataset as groundtruth information to train our student estimator. We considered both Graph-based and Transformer-based architectures as our teacher and student estimators, including the architectures, explained in Section IV.

One caveat with the Transformer-based approach is the lack of 'Inductive Bias', as stated by Dosovitskiy et al. [25], whereby, only MLP layers are local and translationally equivariant, while the self-attention layers are global which doesn't model the two-dimensional neighborhood structure of images that Convolutional Neural Nets [38] extend into each layer of their architecture. This is overcome by pre-training Transformers on a large corpus of data and fine-tuning on a smaller corpus, which has proven to surpass vanilla convolution-based approaches.

To model the Student network, based on our experiments, we found that the Transformer-based architecture has a better computational complexity, while the Graph Convolution-based architecture outperforms in precision (MPJPE). This presents a unique scenario of trade-offs, where both factors tend to be integral for a holistic 3D Pose prediction model. Our goal is to design a student architecture whose objective is to outperform both vanilla Transformer-based and vanilla Graph Convolution-based architectures. To this extent, we propose a hybrid architecture (Trans-Conv), which aims to

extract the best from both, by the introduction of a 3-layer Temporal Convolutional block in parallel to the 3-layer Strided Transformer block in [31]. This is a two-pronged solution to both achieving an ensemble model and inducing inductive bias into our Student Network.

'Trans-Conv' introduces 'sliced convolutions' to the regular Attention-FeedForward Transformer architecture. Inspired by [5], we propose a 3-block Convolutional layer as shown in Figure 3 which shares features with Strided Transformer in [31]. Each block consists of two one-dimensional convolution layers followed by Batch Normalization and Relu activation. To match the dimensions of the output tensors from these blocks, we apply Maxpooling along the residual connection.

We train this network on Human3.6M dataset, following the training scheme as outlined in [31] and [5], training on 5 subjects (S1, S5, S6, S7, S8) and testing on 2 subjects (S9, S11). Both the Transformer and the Convolutional layer are trained with shared weights, as this was determined empirically to improve the performance of the network, rather than a separate training scheme. We found better results using the AdamW optimizer with Amsgrad, training with an initial learning rate of 1e-3, with a 0.95 decay every epoch and a 0.5 decay every 5th epoch, as followed in [31]. We train for 50 epochs, followed by a refinement scheme as adopted in [20]. Finally, the prediction maps from both the transformer layer and the convolutional layer are passed together into a pose regression head to regress the predicted keypoints.

For 3D pose estimation, the default loss metric followed across the literature is Mean Per Joint Position Error (MPJPE), also called Protocol 1, which we adopt in our work. We capture the single target frame scale loss L_c, which minimizes the distance between estimated 3D pose $X \in \mathbb{R}^{J\times 3}$ and the ground truth $Y \in \mathbb{R}^{J\times 3}$, as shown in Equation 2.

$$L_t = \sum_{i=1}^{J} ||Y_i - X_i||_2^{TCN} \tag{2}$$

Since our proposed network is an additional layer added to the existing architecture in [12], we model the total loss as the sum of losses from VTE, STE, and our proposed TCN layer, as shown in equation 3.

$$L = \lambda_v L_v + \lambda_s L_s + \lambda_t L_t \tag{3}$$

In equation 3, $\lambda_v, \lambda_s, \lambda_t$ corresponds to the weighing factors. Once the 3D pose is obtained in the student estimator, the obtained 3D pose keypoints were fed into a pose refinement module. 3D poses can be represented in two ways- as the root coordinates (x', y') obtained from the estimator or by taking the image coordinates and concatenating them with the average depth values from the 3D pose estimator. Both these representations have advantages and disadvantages, for example, the latter representation heavily depends on the 2D pose estimated from the 2D estimator. Therefore, we have captured both representations and taken a weighted average of these representations, and the resultant output is our final 3D refined pose.

VI. Dataset

A. H3.6M Dataset

Human3.6M dataset [37] is, to the best of our knowledge, currently the largest publicly available dataset for human 3D pose estimation. The dataset consists of 3.6 million images featuring 7 professional actors performing 15 everyday activities such as walking, eating, sitting, making a phone call, and engaging in a discussion. 2D joint locations and 3D groundtruth positions are available, as well as projection (camera) parameters and body proportions for all the actors.

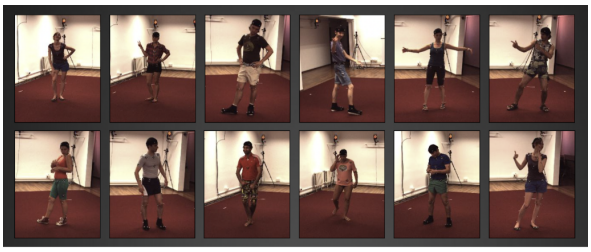


Fig. 4. H3.6M dataset

B. Ice-hockey Dataset

The dataset consists of broadcast video sequences and their corresponding 2D pose estimation data. Some of the frames obtained from the broadcast video of the hockey dataset can be visualized in figure 5.

Fig. 5. Ice-hockey dataset

The dataset consists of a total of 10 games, each sequence recorded at 30 fps and encompasses a total of 9000 frames. Each frame in the dataset is manually annotated with 17 keypoints per player.

VII. Metrics

A. Mean Per Joint Position Error

Mean Per Joint Position Error (MPJPE) is a metric that is widely used for evaluation of pose estimation, which is the L2 distance averaged over all joints. The MPJPE metric is the root-relative Euclidean error averaged over all joints and poses. In a relative root pose, the hip (the root joint) is positioned at the origin. In the evaluation process, we have used MPJPE as a metric for 3D pose estimation.

B. Custom Metric Name

We used the PCK metric to find out the accuracy of our models. We essentially find out the Manhattan distance between the predicted and ground-truth keypoints, and check

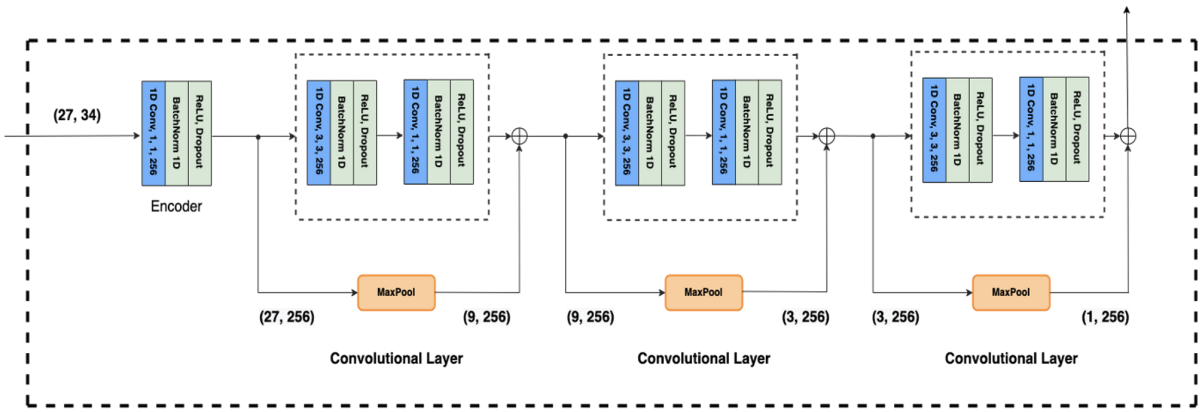
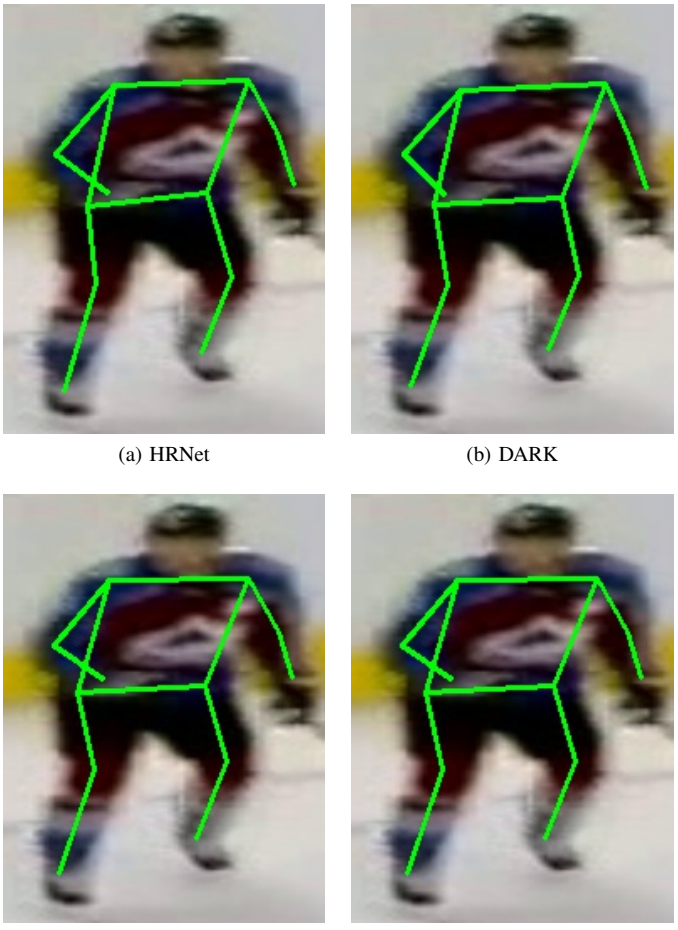


Fig. 3. Novel teacher estimator network

if it is lesser than a threshold (20 in our case). The threshold value was found by doing a grid search over all values from 1 to 100. Additionally, in order to prevent occlusion from skewing our model's performance, we filter out the occluded points by using the confidence score of each prediction and consider prediction only if confidence of prediction is > 0.6.



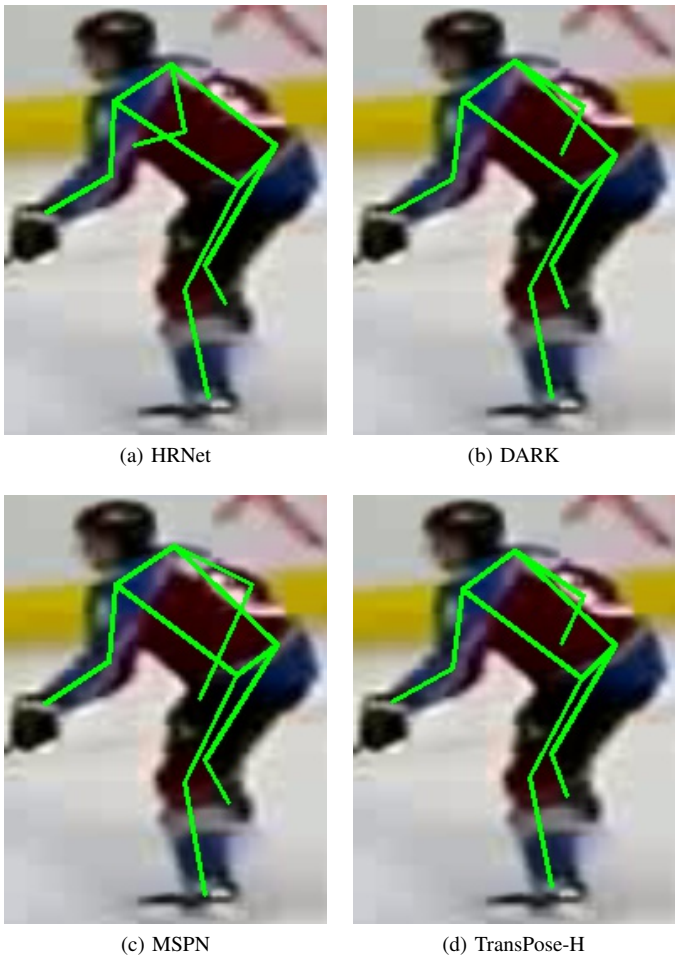
(a) HRNet (b) DARK

(c) MSPN (d) TransPose-H

Fig. 6. Inference of the implemented 2D pose estimators without Occlusion

VIII. Experimentation

A. 2D Pose Estimation



(a) HRNet (b) DARK

(c) MSPN (d) TransPose-H

Fig. 7. Inference of the implemented 2D pose estimators with Occlusion

All of our experiments were conducted using an NVIDIA Geforce RTX 2070 GPU with 8 gigs of RAM. All the 2D pose estimation models were fine-tuned using the pretrained COCO

[39] models for 10 epochs. Because of memory and hardware constraints, we were unable to perform extensive experiments and could only use a batch size of 8. The input size of all the models was (192, 256).

In order to perform all our experiments, we used a 2-stage MSPN, HRNet-W48, DARK with HRnet-W32 as the backbone and TransPose with HRNet-W48 as the feature extractor. Furthermore, for MSPN, we used SGD [40] as our optimizer with a learning rate of 1e-2, momentum of 0.9, and a weight decay of 1e-5. As for all the other models, we used Adam [41] as our optimizer with a learning rate of 1e-3.

Table I shows the training and validation per joint accuracy obtained from the network trained from scratch.

B. 3D Pose Estimation

Considering we don't have groundtruth data for the 3D joints of the players, we implemented a couple of 3D pose estimation algorithms trained on the H36M dataset [37]. We have tested a couple of 3D poses estimation approaches to finalize which estimator can act as a teacher and student. Some of the qualitative experimentation results are shown below for the implemented 3D pose estimators.

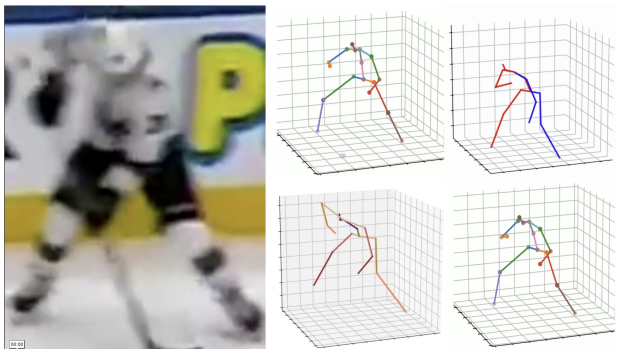
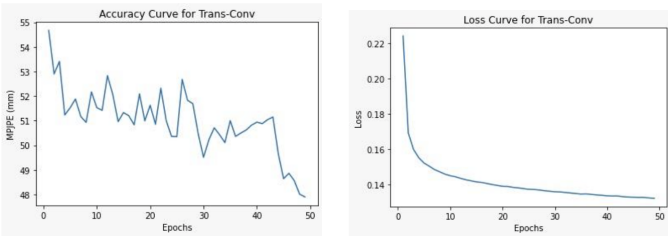


Fig. 9. Inference of the implemented 3D pose estimators

In Figure 9, the left image corresponds to the input image for which inference was done using various 3D pose estimators. The center-top image corresponds to the novel "Trans-Conv" network, right-top corresponds to MHFormer, center-bottom corresponds to GASTNet and the right-bottom corresponds to the STE network.



(a) MPJPE Curve (b) Validation Loss

Fig. 10. MPJPE Curve and loss of the novel architecture

TABLE II
TOTAL AND PER JOINT ACCURACY OF THE FINE-TUNED NETWORKS

Model	Receptive Field	MPJPE (mm)
Hossain et al. [2]	27 frames	58.3
Cai et al. [3]	27 frames	48.8
Pavllo et al. [4]	27 frames	48.6
Chen et al. [5]	27 frames	48.3
Wenhao et al. [1]	27 frames	46.9
Ours (Trans-Conv)	27 frames	**47.9**

Based on the experimentations conducted on various 3D pose estimators, we have observed that graph attention blocks combined with convolution layers model long-range temporal dependencies, thereby handling occlusions and increasing the robustness of our ground truths better when compared to other pose estimators. Thus, we have decided to use GASTNet as the teacher 3D pose estimator considering its robustness to occlusions, and have used the novel proposed network as the student 3D pose estimator.

IX. CONCLUSION

In this work, a novel approach for 3D pose estimation for ice-hockey players was proposed. A teacher-student estimator is used for 3D pose estimation which uses the pose information from the 2D pose estimator along with the input broadcast videos as input. Several 2D pose estimation approaches were fine-tuned and analyzed with our custom ice-hockey dataset. MSPN was recognized to be the most robust performing 2D pose architecture and thus was leveraged as the 2D pose estimator of our final architecture. GASTNet was used as the teacher network considering its robustness to occlusions (which is one of the most common problems in sports like ice-hockey). For the student estimator, we have proposed a novel architecture 'Trans-Conv' network which incorporates sliced convolutions to the regular Attention-FeedForward transformer architecture. The proposed network solves the inductive bias problem in the transformers. Pose refinement on the output 3D pose data from the teacher-student estimator was done to refine the output data.

X. FUTURE WORKS

Through this work, we present multiple open-ended developments in both 2D and 3D pose estimation modeling. In 2D space, further works may explore using graphical models, transformer models with tuned self-attentions, and reinforcement learning-based approach to name a few. Our work can also be extrapolated into game theory by training adversarial agents to generate new ways of tracking poses. In 3D space, our novel architecture can be further hyperparameter-tuned for SOTA performance. Further, the 3D model can also be fine-tuned with 3D groundtruth keypoints, if they are made available for ice-hockey in the future.

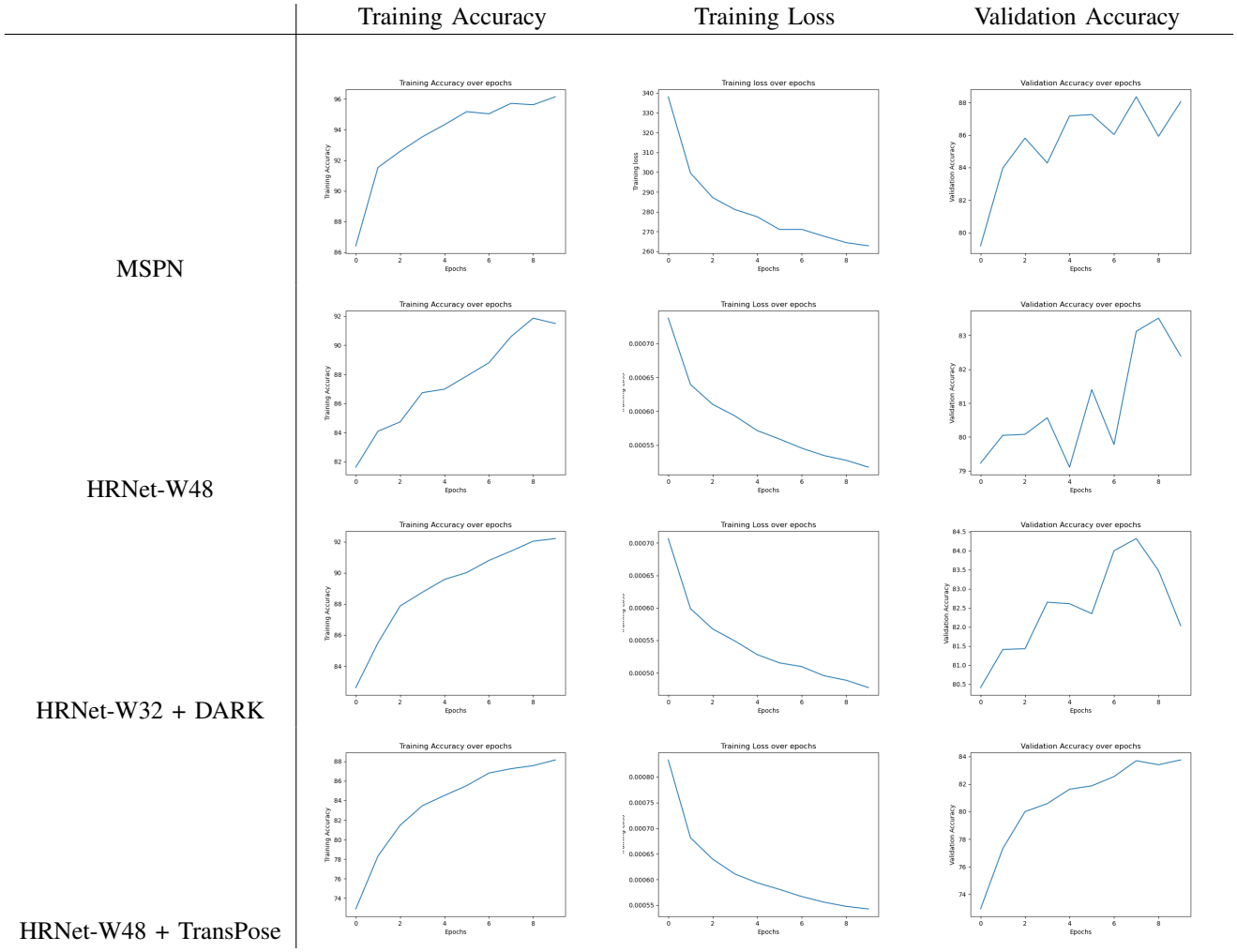


Fig. 8. Training and Validation results of accuracy and overall loss of the fine-tuned 2D pose estimator with our ice-hockey dataset.

TABLE I
TOTAL AND PER JOINT ACCURACY OF THE FINE-TUNED NETWORKS

Joint	Training Accuracy(%)				Validation Accuracy(%)			
	MSPN	HRNet-W48	HRNet-W32 + DARK	TransPose	MSPN	HRNet-W48	HRNet-W32 + DARK	TransPose
Left Shoulder	**97.1**	95.27	95.08	91.44	**90.13**	86.13	86.92	85.83
Right Shoulder	**96.63**	95.23	95.07	91.53	**89.96**	86.23	86.98	84.95
Left Elbow	**94.86**	89.97	90.80	86.92	**87.12**	81.07	82.51	81.95
Right Elbow	**94.91**	88.33	89.52	85.53	**89.03**	81.69	85.96	84.00
Left Wrist	**93.34**	87.69	87.13	83.70	**86.13**	80.27	80.38	80.77
Right Wrist	**92.98**	86.11	84.58	81.16	**86.61**	81.18	82.26	82.04
Left Hip	**93.76**	90.52	89.07	84.78	**79.38**	75.69	75.04	74.80
Right Hip	**94.27**	90.72	88.88	84.4	**79.75**	75.63	78.45	79.50
Left Knee	**97.19**	94.38	94.70	92.59	**92.39**	89.54	89.25	88.27
Right Knee	**97.28**	94.67	94.78	92.15	**92.76**	89.35	90.66	80.81
Left Ankle	**97.21**	93.89	93.02	91.23	**92.01**	87.00	86.51	86.65
Right Ankle	**97.40**	94.18	93.92	91.4	**93.70**	87.48	86.19	86.93
Total accuracy	**95.71**	91.86	91.40	83.75	**88.35**	83.51	84.32	88.16

REFERENCES

[1] Wenbo Li, Zhicheng Wang, Binyi Yin, Qixiang Peng, Yuming Du, Tianzi Xiao, Gang Yu, Hongtao Lu, Yichen Wei, and Jian Sun. Rethinking on multi-stage networks for human pose estimation. *arXiv preprint arXiv:1901.00148*, 2019.

[2] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, 2019.

[3] Feng Zhang, Xiatian Zhu, Hanbin Dai, Mao Ye, and Ce Zhu. Distribution-aware coordinate representation for human pose estimation. *CoRR*, abs/1910.06278, 2019.

[4] Sen Yang, Zhibin Quan, Mu Nie, and Wankou Yang. Transpose: Towards explainable human pose estimation by transformer. *CoRR*, abs/2012.14214, 2020.

[5] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[6] Vasileios Belagiannis, Sikandar Amin, Mykhaylo Andriluka, Bernt Schiele, Nassir Navab, and Slobodan Ilic. 3d pictorial structures for multiple human pose estimation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1669–1676, 2014.

[7] Yi Yang and Deva Ramanan. Articulated human detection with flexible mixtures of parts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2878–2890, 2013.

[8] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893 vol. 1, 2005.

[9] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. *CoRR*, abs/1312.4659, 2013.

[10] Leonid Pishchulin, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter V. Gehler, and Bernt Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. *CoRR*, abs/1511.06645, 2015.

[11] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. *CoRR*, abs/1603.06937, 2016.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[13] Junjie Huang, Zheng Zhu, Feng Guo, and Guan Huang. The devil is in the details: Delving into unbiased data processing for human pose estimation. *CoRR*, abs/1911.07524, 2019.

[14] Yuanhao Cai, Zhicheng Wang, Zhengxiong Luo, Binyi Yin, Angang Du, Haoqian Wang, Xinyu Zhou, Erjin Zhou, Xiangyu Zhang, and Jian Sun. Learning delicate local representations for multi-person pose estimation. *CoRR*, abs/2003.04030, 2020.

[15] Yanjie Li, Shoukui Zhang, Zhicheng Wang, Sen Yang, Wankou Yang, Shu-Tao Xia, and Erjin Zhou. Tokenpose: Learning keypoint tokens for human pose estimation. *CoRR*, abs/2104.03516, 2021.

[16] Ke Li, Shijie Wang, Xiang Zhang, Yifan Xu, Weijian Xu, and Zhuowen Tu. Pose recognition with cascade transformers. *CoRR*, abs/2104.06976, 2021.

[17] Yuhui Yuan, Rao Fu, Lang Huang, Weihong Lin, Chao Zhang, Xilin Chen, and Jingdong Wang. Hrformer: High-resolution transformer for dense prediction. *CoRR*, abs/2110.09408, 2021.

[18] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation, 2022.

[19] Muhammed Kocabas, Salih Karagoz, and Emre Akbas. Self-supervised learning of 3d human pose using multi-view geometry. *CoRR*, abs/1903.02330, 2019.

[20] Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional neural network: A deep learning framework for traffic forecasting. *CoRR*, abs/1709.04875, 2017.

[21] Junfa Liu, Juan Rojas, Zhijun Liang, Yihui Li, and Yisheng Guan. A graph attention spatio-temporal convolutional networks for 3d human pose estimation in video. *arXiv preprint arXiv:2003.14179*, 2020.

[22] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[23] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *CoRR*, abs/2005.14165, 2020.

[24] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. *CoRR*, abs/2102.12092, 2021.

[25] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020.

[26] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. *CoRR*, abs/2005.12872, 2020.

[27] Yunhe Gao, Mu Zhou, Di Liu, Zhennan Yan, Shaoting Zhang, and Dimitris N. Metaxas. A data-scalable transformer for medical image segmentation: Architecture, model efficiency, and benchmark, 2022.

[28] Fangao Zeng, Bin Dong, Tiancai Wang, Cheng Chen, Xiangyu Zhang, and Yichen Wei. MOTR: end-to-end multiple-object tracking with transformer. *CoRR*, abs/2105.03247, 2021.

[29] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixé, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. *CoRR*, abs/2101.02702, 2021.

[30] Xingyi Zhou, Tianwei Yin, Vladlen Koltun, and Philipp Krähenbühl. Global tracking transformers, 2022.

[31] Wenhao Li, Hong Liu, Runwei Ding, Mengyuan Liu, and Pichao Wang. Lifting transformer for 3d human pose estimation in video. *CoRR*, abs/2103.14304, 2021.

[32] Wenhao Li, Hong Liu, Hao Tang, Pichao Wang, and Luc Van Gool. Mhformer: Multi-hypothesis transformer for 3d human pose estimation. *CoRR*, abs/2111.12707, 2021.

[33] Helmut Neher, Mehrnaz Fani, David A Clausi, Alexander Wong, and John S. Zelek. Pose estimation of players in hockey videos using convolutional neural networks. 2017.

[34] Mehrnaz Fani, Helmut Neher, David A. Clausi, Alexander Wong, and John Zelek. Hockey action recognition via integrated stacked hourglass network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 85–93, 2017.

[35] Helmut Neher, Kanav Vats, Alexander Wong, and David Clausi. Hyperstacknet: A hyper stacked hourglass deep convolutional neural network architecture for joint player and stick pose estimation in hockey. pages 313–320, 05 2018.

[36] William J. McNally, Kanav Vats, Alexander Wong, and John McPhee. Evopose2d: Pushing the boundaries of 2d human pose estimation using neuroevolution. *CoRR*, abs/2011.08446, 2020.

[37] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, jul 2014.

[38] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551, 1989.

[39] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Doll'a r, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014.

[40] Sebastian Ruder. An overview of gradient descent optimization algorithms. *CoRR*, abs/1609.04747, 2016.

[41] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.