3D Human Pose Lifting with Grid Convolution

Yangyuxuan Kang[1,2*], **Yuyang Liu**[3*], **Anbang Yao**[4†], **Shandong Wang**[4], **Enhua Wu**[1,2,5†]

[1] State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences
[2] University of Chinese Academy of Sciences
[3] Tsinghua University
[4] Intel Labs China
[5] Faculty of Science and Technology, University of Macau
kyyx@ios.ac.cn, yyliu22@mails.tsinghua.edu.cn, {anbang.yao,shandong.wang}@intel.com, ehwu@um.edu.mo

Abstract

Existing lifting networks for regressing 3D human poses from 2D single-view poses are typically constructed with linear layers based on graph-structured representation learning. In sharp contrast to them, this paper presents Grid Convolution (GridConv), mimicking the wisdom of regular convolution operations in image space. GridConv is based on a novel Semantic Grid Transformation (SGT) which leverages a binary assignment matrix to map the irregular graph-structured human pose onto a regular weave-like grid pose representation joint by joint, enabling layer-wise feature learning with GridConv operations. We provide two ways to implement SGT, including handcrafted and learnable designs. Surprisingly, both designs turn out to achieve promising results and the learnable one is better, demonstrating the great potential of this new lifting representation learning formulation. To improve the ability of GridConv to encode contextual cues, we introduce an attention module over the convolutional kernel, making grid convolution operations input-dependent, spatial-aware and grid-specific. We show that our fully convolutional grid lifting network outperforms state-of-the-art methods with noticeable margins under (1) conventional evaluation on Human3.6M and (2) cross-evaluation on MPI-INF-3DHP. Code is available at https://github.com/OSVAI/GridConv.

Introduction

3D human pose estimation is essential for various applications. The task aims to recover the 3D positions of human body joints from images or videos. Benefiting from great advances in deep learning techniques, 3D human pose estimation with a single image input has now become practical.

One mainstream solution family estimates 3D human pose in two stages. The first stage detects 2D pose in an image, and the second stage lifts detected 2D pose to its 3D estimate. Along with the advent of many well-designed 2D pose detectors, such as HRNet (Sun et al. 2019), 2D human pose detection technology is gradually becoming mature, showing significantly improved performance, even in outdoor scenarios with dramatic changes of background and rarely seen situations with diverse occlusions. Driven by this as well as the prevalence of effective methods to generate large amounts of 2D-to-3D human pose pairs, 2D-to-3D pose lifting has become a critical research topic, and thus has attracted increased attention recently. Many works (Fang et al. 2018; Zhao et al. 2019; Kang et al. 2020) have been devoted to advancing 2D-to-3D pose lifting research. These works typically represent human pose as a 1D feature vector or a *graph*, and use either fully connected network or graph convolutional network to regress 3D pose from 2D input.

However, we observe that a pretty successful family of deep learning techniques, convolutional neural networks for image recognition and editing tasks, does not attract the interest of researchers in the lifting field. A vital reason is that graph-structured human skeleton pose having unbalanced joint neighborhoods hinders the use of convolution operation with regular kernels. Motivated by the observation, we address the pose lifting problem by formulating a novel grid-based representation learning paradigm, attempting to introduce a 2D coordinate system to measure joint relationships and enable regular convolution operations and advanced design of building blocks. Regarding our goal, three critical questions need to be considered: (1) Is it possible to transform a human skeleton pose into an image-like grid coordinate system? (2) How to preserve intrinsic joint relationships of human skeleton during the transformation? (3) After such transformation, can we use few modifications to convolutional networks to pursue a high-performance lifting model?

To the first question, we propose *Semantic Grid Transformation* (SGT) which maps the irregular graph-structured human pose onto a regular *weave-like grid pose representation* joint by joint. To the second question, we design a handcrafted layout that meanwhile preserves skeleton topology and brings in a new kind of motion semantics. In addition, to explore a better grid layout, we propose a learning-based SGT called *AutoGrids* that automatically searches the layout conditioned on the input distribution. To the last question, SGT enables a new type of standard convolution operations on the grid pose. We call this operation paradigm *Grid Convolution* (GridConv). We further enhance the learning capability of GridConv by introducing an attention mod-

[*] This work was done when they were interns at Intel Labs China, supervised by Anbang Yao.

[†] Corresponding authors.

ule over the convolutional kernels, making GridConv input-dependent, spatial-aware and grid-specific.

The solutions for the above three questions constitute our core contributions to constructing a new category of fully convolutional network that lifts 2D pose to 3D estimate in the weave-like grid pose domain. Extensive experiments on public 3D human pose estimation datasets demonstrate superior performance of our fully convolutional lifting network to existing methods by using the proposed representation learning paradigm. Furthermore, our method retains its effectiveness in the augmented training regime with synthetic data or joint optimization of the 2D human pose detector and the 2D-to-3D lifting network, showing further improved performance.

Related Work

End-to-End 3D Human Pose Estimation

In the pre-deep-learning era, 3D human pose estimation usually relies on building a 2D pictorial model from images and inferring plausible 3D targets from 2D evidence by Bayesian probabilistic models (Belagiannis et al. 2014; Andriluka, Roth, and Schiele 2010). With the booming of deep learning techniques and the availability of high-quality 3D human dataset in the community, tremendous progress has been achieved by end-to-end learning of 3D human pose estimation (Pavlakos et al. 2017; Mehta et al. 2017b; Zhou et al. 2017, 2021). These approaches show significant advantages over traditional ones.

With the rise of convolutional neural network techniques, the technology for a related task, namely 2D human pose detection, has become more and more mature. In recent years, many prevailing 2D human pose detectors, such as OpenPose (Cao et al. 2019) and HRNet (Sun et al. 2019), have been proposed. Under this context, one critical research problem arises: it is possible to infer 3D human pose directly from 2D pose detection? To this problem, an early work (Zhou et al. 2016) used sparse representation on 3D pose and inferred 3D pose under the condition of giving 2D pose probability heatmap or coordinate. Later on, (Martinez et al. 2017) proposed the two-stage 3D human pose estimation paradigm with deep learning, which first detects 2D body keypoints in an image and then regresses 3D pose coordinate from 2D pose coordinate. Since then, a lot of methods have been proposed to improve 2D-to-3D pose lifting scheme.

2D-to-3D Pose Lifting

The pioneering lifting work (Martinez et al. 2017) treated input 2D pose as a generic 1D feature vector and directly regressed 3D joint coordinate. Subsequent works attempted to leverage prior knowledge of the human body skeleton to improve lifting optimization. For example, (Sun et al. 2017) exploited joint connection structure by representing pose as a composition of bones. (Dabral et al. 2018) introduced illegal articulation angle penalty and body symmetry constraint in the training process. (Fang et al. 2018) modeled skeleton motion in three high-level aspects including kinematics, symmetry, and motor coordination by defining joint relations in a recurrent network. Our work is similar to them in modeling high-level joint relations, but the proposed representation learning paradigm is differentiated from the others.

With the arising research of Graph Convolutional Networks (GCNs), many works represented human pose as a graph by mapping joints and limbs as graph nodes and edges, and substituted fully connected networks by GCNs as their lifting models. Most of them (Ci et al. 2019; Zhao et al. 2019; Cai et al. 2019; Liu et al. 2020; Zou et al. 2020) focused on developing pose-relevant graph convolution operators and network architectures. Some works (Zeng et al. 2021; Hu et al. 2021) argued that the default skeletal graph is sub-optimal for perceiving long-distance joint relations, and thus proposed to dynamically adjust the graph structure.

This work goes beyond graph representation for human pose and formulates a semantically more informative lifting representation learning paradigm. Moreover, with the help of some sophisticated strategies that generate large-scale synthetic 2D-to-3D data (Gong, Zhang, and Feng 2021), our method can get further improvement after fine-tuning with augmented data, showing its great generalization ability.

Method

In this section, we first describe the formulation of Semantic Grid Transformation (SGT), which maps graph-structured human pose to a uniform weave-like grid pose representation, giving birth to Grid Convolution (GridConv). For SGT, we propose a handcrafted design and a learnable one. Then we present the concept of GridConv as well as its dynamic form. Finally, we detail the architecture of our grid lifting network shown in Figure 1.

Semantic Grid Transformation

Suppose that we have a human pose $G \in \mathbb{R}^{J \times C}$, where J denotes the number of body joints, C denotes the coordinate dimensions for each joint ($C = 2$ for 2D pose, and $C = 3$ for 3D pose). The basic goal of SGT is to construct a *grid pose* $D \in \mathbb{R}^{H \times P \times C}$, defined as a regular weave-like grid representation with the spatial size of H×P filled by J body joints, where $HP{\geq}J$. By changing the setting of H×P, a grid pose D could be either square or rectangular in shape, e.g., 5×5 or 7×3. SGT mapping function ϕ is defined as:

$$D = \phi(G) = S \times G, \tag{1}$$

where $S \in \{0, 1\}^{HP \times J}$ is a binary assignment matrix which maps the graph-structured human pose G to the desired grid pose D joint by joint. During the mapping, a grid node $D_p, p \in [1, HP]$ in the grid pose D will be filled by a particular body joint $G_j, j \in [1, J]$ only if $S_{p,j} = 1$. Inverse SGT ϕ^{-1} that maps D to G is formed by inversing the assignment process.

Recall that existing lifting methods typically adopt skeleton graph as pose representation. When constructing a grid pose D, it is natural to allow the desired grid pose to inherit joint features and preserve skeleton topology. Concretely, given an edge set of skeleton graph $\boldsymbol{E}$, this goal can be ac-

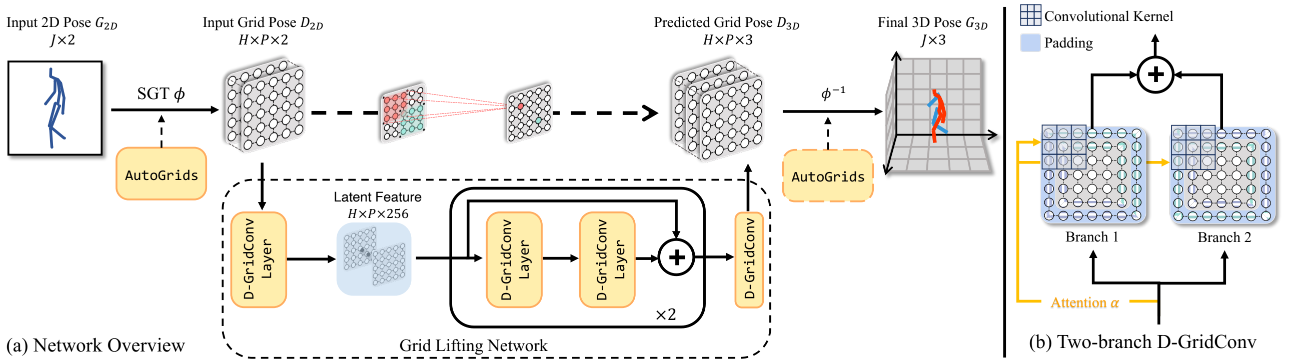


Figure 1: (a) An architectural overview of the proposed grid lifting network. Here, 2D grid pose D_{2D}, namely the network input, is transformed from 2D human pose input G_{2D} via an SGT module ϕ. D-GridConv layers learn latent feature embedding on the grid pose. At the end of the network, an inverse SGT module ϕ^{-1} rearranges 3D grid pose D_{3D} to target 3D human pose G_{3D}. (b) The internal architecture of D-GridConv module. The output grid pose is obtained by summing up two-branched convolution results of padded inputs.

complished by adding the following two constraints to ϕ:

$$S_{p,i} \times \sum_{q\in \boldsymbol{N}(p)} S_{q,j} \geq 1, \exists p \in [1, HP], \qquad \forall (i,j) \in \boldsymbol{E} \quad (2)$$

$$\sum_{k=1}^{J} S_{p,k} = 1, \quad \forall p \in [1, HP], (3)$$

where $\boldsymbol{N}(\cdot)$ denotes the neighborhood of a certain grid node, namely four adjacent nodes in the horizontal and vertical directions.

On the one hand, by satisfying Equation (2), originally connected body joints remain adjacent in the resulting grid pose. On the other hand, Equation (3) restricts each row of S as a one-hot vector, which allows each grid node to have explicit semantic meaning (coordinate of a specific joint). Equation (2) and (3) produce replicants of some joints in the resulting grid layout and provide a loose collection of solutions. This formulation of SGT earns two merits for incubating handcrafted design and the learnable one.

Merits of SGT. (1) Grid nodes having both vertical and horizontal edges offer multiple connection types for depicting joint relationships, which allows us to handcraft a semantically richer pose structure. (2) A large number of solutions existing in the assignment space make it possible to define a learnable SGT by first describing the space in continuous distribution and then searching an optimal point.

Two Designs for Implementing SGT

In light of the above discussion, we define and analyze the advantages of SGT. Next, we provide a handcrafted SGT as a basic design. And then, we present AutoGrids that automatically learns SGT as a generalized design.

Handcrafted SGT design. We heuristically make the resulting grid pose well encode both the vertical (along kinematic forward direction) and the horizontal (along kinematic peer direction) relationships of body joints to the root joint (e.g., torso joint), preserving prior joint connections of the skeleton pose graph structure. The corresponding grid layout is shown in Figure 2. Some joints have replicants in the grid, which are averaged during inverse SGT. In Section 4, we test the efficacy of such a handcrafted layout and compare it with many other variants.

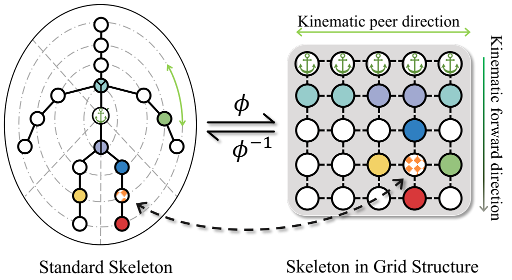


Figure 2: Handcrafted SGT. In a heuristic manner, torso joint is set as the anchor. The remaining joints are arranged along kinematic forward and peer directions into the vertical and horizontal directions of the grid structure.

Although handcrafted SGT already achieves remarkable performance, it still faces some issues, such as the scenario using a new definition of skeleton, where redesigning of SGT is required. It motivates us to seek an automatic formulation and to further excavate the learning potential of grid representation.

Learnable SGT design. We shelve the constraint on preserving prior graph-structured joint connections defined in Equation (2), and propose a learnable module called *AutoGrids* to learn an adaptive assignment matrix conditioned on the input human skeleton pose, which is jointly optimized with our lifting network (its architecture will be clarified later).

To learn an assignment matrix S filled by discrete binary values, the difficulty lies in how to make the training process differentiable. To address this problem, we adopt Gumbel Softmax (Jang, Gu, and Poole 2017) which uses a continuous distribution of assignment matrix to approximate the sampling of S. Let $S^{prob} \in \mathbb{R}^{HP\times J}$ be a probability distribution of an assignment matrix filled by continuous positive values, whose element S^{prob}_{ij} indicates the probability score assigning joint G_j of skeleton pose graph to grid node D_i.

During training, AutoGrids module generates a soft as-

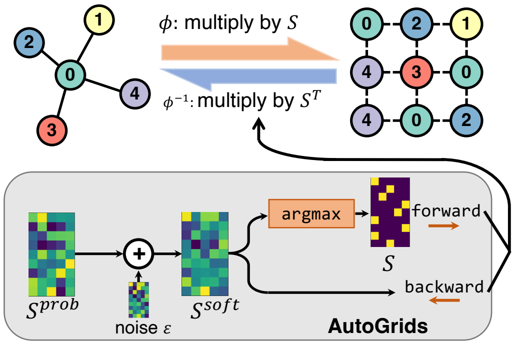


Figure 3: Illustration of the learning process of AutoGrids.

signment matrix S^{soft} by:

$$S^{soft} = S^{prob} + \varepsilon, \tag{4}$$

where $\varepsilon \in \mathbb{R}^{HP \times J}$ is a Gumbel noise that assists to resample the soft assignment matrix S^{soft} from the probability distribution S^{prob}. For forward inference, the desired binary assignment matrix S can be easily determined by taking the highest probability response per row on S^{soft} according to:

$$S_i = onehot\big(\arg\max_j S_{ij}^{soft}\big). \tag{5}$$

This discretization operation cuts off the backward gradient flow during training, so we use straight-through estimator (Courbariaux, Bengio, and David 2015) for parameter update. Specifically, in the backward, continuous gradient approximation is used to directly update S^{soft}.

Introduced noise interference encourages the exploration on different grid pose proposals, which facilitates AutoGrids module to identify a decent grid layout. Figure 3 illustrates the learning process of AutoGrids. In the implementation, AutoGrids module is jointly trained with the lifting network in an end-to-end manner. Experiments in Section 4 show that the learnable SGT works better than the handcrafted design. And promising results of two designs validate the great potential of grid representation learning paradigm.

Grid Convolution and Its Dynamic Form

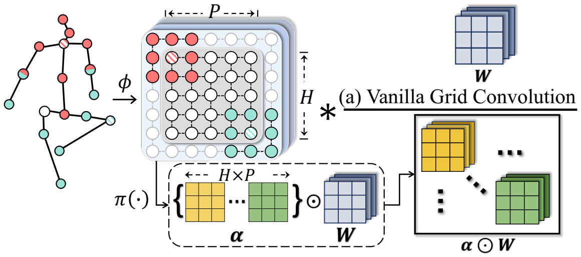
Given a constructed grid pose D, now we can easily define convolution operation on grid pose, dubbed *GridConv*, resembling regular convolution operations in image space.

Vanilla form of GridConv. Mathematically, standard GridConv operation is defined as:

$$D^{out} = W * D^{in}, \tag{6}$$

where $*$ denotes the convolution operation; $D^{in} \in \mathbb{R}^{H \times P \times C^{in}}$ and $D^{out} \in \mathbb{R}^{H \times P \times C^{out}}$ denote the input feature and the output feature, respectively; $W \in \mathbb{R}^{K \times K \times C^{in} \times C^{out}}$ denotes the convolutional kernel with kernel size K×K. With proper padding strategy, the spatial size H×P is maintained throughout the input and output of a GridConv layer, which sharply contrasts with prevalent convolutional neural networks for image recognition tasks that typically reduce spatial feature size at multiple stages.

Dynamic form of GridConv. According to the above definition, with vanilla GridConv, convolutional kernel W is shared to the input feature, with no consideration of different grid locations or diverse body motions. To strengthen its feature learning ability on rich contextual cues, we leverage the attention mechanism conditioned on the input feature to generate attentive scaling factors to adjust the convolutional kernel, making grid convolution operations input-dependent, spatial-aware and grid-specific. We call this variant *Dynamic Grid Convolution (D-GridConv)*. Specifically, following Equation (6), D-GridConv is defined as:



(b) Dynamic Grid Convolution

Figure 4: Illustration of (a) Vanilla GridConv and (b) Dynamic GridConv. For simplicity, just show a single filter of the convolutional kernel.

$$\alpha = \pi(D^{in}) \tag{7}$$

$$D_{ij}^{out} = (\alpha_{ij} \odot W) * D_{\delta_{ij}}^{in}, \tag{8}$$

where π denotes the attention module (defined as an SE-typed structure (Hu, Shen, and Sun 2018)) to generate the input-dependent scaling factor $\alpha \in \mathbb{R}^{H \times P \times K \times K}$ for adjusting the convolutional kernel W. Specifically, W is multiplied by $\alpha_{ij} \in \mathbb{R}^{K \times K}$ on each grid patch in an element-wise manner across channel dimension. δ_{ij} denotes the index vector of local grid patch centered on grid (i, j) where $i \in [1, H], j \in [1, P]$. Figure 4a and 4b respectively illustrate how vanilla GridConv acts and how the attentive factor makes D-GridConv dynamically change with respect to grid pose. Custom-designed attention predictor on grid convolution distinguishes D-GridConv from the series of existing attention methods.

Grid Lifting Network

With the above two components, SGT and D-GridConv, now we can construct a new category of fully convolutional lifting network in the grid pose domain, which we call *Grid Lifting Network* (GLN). We use ϕ to map either detected or labeled 2D pose G_{2D} to 2D grid pose D_{2D} as the input to GLN. Then GLN uses a D-GridConv layer to expand the channel dimension of the 2D grid pose from 2 to 256, next uses two residual blocks (each incorporates two D-GridConv layers and a skip connection) to learn latent feature embedding progressively, and then uses another D-GridConv layer to shrink the channel dimension from 256 to 3, and finally gets 3D pose estimate G_{3D} by applying ϕ^{-1} over the 3D output from the last D-GridConv layer. Figure 1 shows an architectural overview of our GLN. The entire processing pipeline of our GLN takes the following form:

$$G_{3D} = \phi^{-1}\left(\boldsymbol{GLN}\left(\phi\left(G_{2D}\right)\right)\right). \tag{9}$$

GLN is trained by minimizing the L_2-norm distance between the inverse 3D pose in graph structure $G_{_{3D}}$ and the ground truth pose $G^{^{GT}}_{_{3D}}$ over all training samples:

$$\mathcal{L} = ||G_{_{3D}} - G^{^{GT}}_{_{3D}}||^{2}_{2}. \tag{10}$$

Relationship with GCN

The proposed grid-structured representation learning paradigm mainly differs from GCN in two aspects: (1) *Data structure*. Grid pose encodes parent-child relation (along kinematic forward direction) and symmetry relation (along kinematic peer direction) in the bidirectional layout, yet graph pose encodes only the former one. (2) *Feature mapping scheme*. GridConv aggregates all latent channels of the neighboring nodes in a single step, yet GCN separates it into two steps recognized as latent feature mapping and neighborhood aggregation. The single-step scheme allows GridConv to fully exploit relations of latent features.

Experiments

Datasets

Human3.6M. It is the largest indoor 3D human motion benchmark with 3D labels collected by motion capture system (Ionescu et al. 2014). The dataset consists of 11 actors playing a variety of activities. We follow the convention that takes *Subject 1,5,6,7,8* as the training set and *Subject 9,11* as the evaluation set. We measure the result by Mean Per Joint Position Error (MPJPE) in millimeters under three protocols. **Protocol 1 (P1)** takes 2D pose detection from HRNet (Sun et al. 2019) as input. **Protocol 1* (P1*)** takes ground truth 2D pose as input. **Protocol 2 (P2)** takes 2D pose detection as input and measures 3D error after aligning 3D estimate to the ground truth through rigid alignment.

MPI-INF-3DHP. It is another 3D human motion benchmark with 3D labels obtained by multi-view reconstruction (Mehta et al. 2017a). To evaluate the generalization ability of our method, we consider challenging cross-dataset evaluation, applying the model trained on Human3.6M for direct test on the evaluation set of MPI-INF-3DHP. The evaluation metrics include MPJPE, Percentage of Correct Keypoints (PCK), and Area Under the Curve (AUC) of PCK.

Implementation Details

Considering that the number of body joints J popularly used for these two datasets is 17, the size of grid pose H×P should be no smaller than 6×3 or 5×4 due to $HP \geq J$. We use a grid pose with 5×5 size as our default setting.

In the grid lifting network, each D-GridConv layer is composed of two-branch D-GridConv, batch normalization, ReLU, and dropout operations. The attention module of D-GridConv consists of global average pooling followed by batch normalization and ReLU, two linear layers (reducing channel dimension first to 16 and further to 3), and a Sigmoid activation function. The convolutional kernel size is fixed to 3×3. Two-branch D-GridConv divides the feature extraction into two branches, applying grid convolution on circular padded and replicate padded grid pose respectively, and finally outputs the sum of their results.

We train the model with Adam optimizer using a batch size of 200 and a learning rate starting at 0.001 for 100 epochs. In AutoGrids, S^{prob} is initialized by handcrafted SGT for 5×5 grid and by random value for other sizes. We stop adding Gumbel noise at the 30th epoch to slow down the rate of grid pose changes. Our model has totally 4.79 million learnable parameters with 0.04 million from the attention modules and <1k learnable parameters from AutoGrids. We train and test the model on a single NVIDIA 1080Ti GPU. Commonly, one run of model training takes about 40 hours, and the runtime speed is over 1600 FPS.

Comparison with State-of-the-Art Methods

First, we describe experimental comparisons under conventional single-dataset evaluation on Human3.6M and challenging cross-dataset evaluation on MPI-INF-3DHP.

Results on Human3.6M. In the upper half part of Table 1, we compare our method with mainstream lifting methods on Human3.6M. Note that two works (Ci et al. 2019; Zeng et al. 2021) marked by § first predict 2D pixel coordinate and a depth value for each joint, and then post-process them into 3D physical coordinate with given camera intrinsics and body root position in camera space. For a fair comparison with them, we also report our result §, showing remarkable gains ($>$1.6 mm). For those approaches dealing with temporal 2D pose input, we report their single-frame results. We can see that our method achieves the best MPJPE of 47.6 mm compared to existing lifting works. And data normalization strategy § pushes our performance further to 46.3 mm, showing great margins against lifting methods.

In the lower half part of Table 1, we jointly train the 2D detector and our grid lifting network in an end-to-end manner, and compare our method to mainstream end-to-end methods. Joint training improves our method to correct erroneous 2D detection results and shows significant model performance improvements against the lifting-alone training. Besides, our model from joint training outperforms most end-to-end methods and approaches state of the art.

We further evaluate our method under all three protocols, and summarize the performance comparison in Table 2. Generally, our method outperforms most of existing works under 2D ground truth input (P1*). When adopting data normalization strategy §, our method is superior to state of the arts under both 2D detection input and GT input.

Results on 3DHP. To better explore the generalization ability of our method, we compare it with previous works that adopt cross-dataset evaluation. Detailed results are shown in Table 3, in which we also include existing works performing both from-scratch training and evaluation on 3DHP, in order to have a more comprehensive comparison. We can observe that our method outperforms all methods by significant margins with respect to all three metrics.

Qualitative Results

Next, we provide some qualitative comparisons of our best-performed model trained on Human3.6M to illustrate the ability to handle challenging scenarios with various viewpoints and severe occlusions. Figure 5a and 5b show visualization results respectively on Human3.6M and on 3DHP.

Method	Dir	Dis	Eat	Gre	Phon	Phot	Pos	Pur	Sit	SitD	Smok	Wait	Dog	Wal	Tog	Avg
Martinez 2017	51.8	56.2	58.1	59.0	69.5	78.4	55.2	58.1	74.0	94.6	62.3	59.1	65.1	49.5	52.4	62.9
Lee 2018	43.8	51.7	48.8	53.1	52.2	74.9	52.7	44.6	56.9	74.3	56.7	66.4	47.5	68.4	45.6	55.8
Fang 2018	50.1	54.3	57.0	57.1	66.6	73.3	53.4	55.7	72.8	88.6	60.3	57.7	62.7	47.5	50.6	60.4
Zhao 2019	47.3	60.7	51.4	60.5	61.1	**49.9**	47.3	68.1	86.2	**55.0**	67.8	61.0	**42.1**	60.6	45.3	57.6
Pavllo 2019	47.1	50.6	49.0	51.8	53.6	61.4	49.4	47.4	59.3	67.4	52.4	49.5	55.3	39.5	42.7	51.8
Sharma 2019	48.6	54.5	54.2	55.7	62.6	72.0	50.5	54.3	70.0	78.3	58.1	55.4	61.4	45.2	49.7	58.0
Ci 2019 §	46.8	52.3	44.7	50.4	52.9	68.9	49.6	46.4	60.2	78.9	51.2	50.0	54.8	40.4	43.3	52.7
Cai 2019	46.5	48.8	47.6	50.9	52.9	61.3	48.3	45.8	59.2	64.4	51.2	48.4	53.5	39.2	41.2	50.6
Li 2020	47.0	**47.1**	49.3	50.5	53.9	58.5	48.8	45.5	55.2	68.6	50.8	47.5	53.6	42.3	45.6	50.9
Zeng 2020	44.5	48.2	47.1	47.8	51.2	56.8	50.1	45.6	59.9	66.4	52.1	45.3	54.2	39.1	40.3	49.9
Zeng 2021 §	43.1	50.4	**43.9**	45.3	**46.1**	57.0	46.3	47.6	56.3	61.5	47.7	47.4	53.5	**35.4**	**37.3**	47.9
Ours	43.1	47.7	44.8	44.9	50.7	55.1	46.3	**42.6**	53.7	63.9	46.3	45.5	50.1	38.6	40.1	**47.6**
Ours §	**39.9**	47.7	44.7	**43.9**	49.2	53.5	**44.4**	43.7	**53.1**	61.6	**45.4**	**44.7**	47.4	37.7	37.9	**46.3**
Sun 2018	47.5	47.7	49.5	50.2	51.4	55.8	43.8	46.4	58.9	65.7	49.4	47.8	49.0	38.9	43.8	49.6
Yang 2018	51.5	58.9	50.4	57.0	62.1	65.4	49.8	52.7	69.2	85.2	57.4	58.4	43.6	60.1	47.7	58.6
Moon 2019	50.5	55.7	50.1	51.7	53.9	55.9	46.8	50.0	61.9	68.0	52.5	49.9	41.8	56.1	46.9	53.3
Moon 2020	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	55.7
Zhou 2021	**34.4**	**42.4**	**36.6**	**42.1**	**38.2**	**39.8**	**34.7**	**40.2**	**45.6**	60.8	**39.0**	**42.6**	**42.0**	**29.8**	**31.7**	**39.9**
Ours e2e	36.9	44.4	41.9	43.3	45.6	47.8	43.0	40.7	50.7	**60.6**	44.3	43.6	43.9	33.9	35.0	43.7

Table 1: MPJPE comparison (mm) of our method against both mainstream lifting and end-to-end methods on Human3.6M. For the comparison with lifting methods (upper half of the table), we report the results under Protocol 1 using 2D detection input. T=1 denotes single-frame results of temporal methods. § denotes estimating 2D pixel and 3D depth jointly. For the comparison with end-to-end methods (lower half of the table), we report the results under image input.

Method	Special Mark	MPJPE		
		P1	P1*	P2
Martinez 2017	-	62.9	45.5	47.7
Zhao 2019	-	57.6	43.8	-
Fang 2018	-	60.4	-	45.7
Sharma 2019	-	58.0	-	40.9
Pavllo 2019	T=1	51.8	-	40.0
Ci 2019	§	52.7	36.3	42.2
Cai 2019	T=1	50.6	38.1	40.2
Zeng 2020	-	49.9	36.4	-
Li 2020	-	50.9	34.5	38.0
Yu 2021	-	67.0	40.1	-
Gong 2021	16 joints	50.2	36.9	39.1
Zeng 2021	§	47.9	30.4	39.0
Ours	-	47.6	36.4	**37.4**
Ours	§	**46.3**	**29.5**	37.6

Table 2: Comparison on Human3.6M under all protocols. § denotes estimating 2D pixel and 3D depth jointly.

Method	Cross Eval	PCK	AUC	MPJPE
Mehta *et al.* 2017a	✗	76.5	40.8	117.6
Mehta *et al.* 2017b	✗	76.6	40.4	124.7
LCR-Net 2017	✗	59.6	27.6	158.4
Zhou *et al.* 2017	✗	69.2	32.5	137.1
Multi Person 2018	✗	75.2	37.8	122.2
OriNet 2018	✗	81.8	45.2	89.4
HMR 2018	✓	77.1	40.7	113.2
LCN 2019	✓	74.0	36.7	-
Li *et al.* 2020	✓	81.2	46.1	99.7
SRNet 2020	✓	77.6	43.8	-
SkeletalGCN 2021	✓	82.1	46.2	-
PoseAug 2021	✓	88.6	57.3	73.0
Ours	✓	**89.2**	**57.6**	**72.1**

Table 3: Performance comparison on MPI-INF-3DHP.

Method	Original	G_{craft}	D_{craft}
SemGCN (2019)	43.8	41.8	43.5
LCN (2019)	39.7	39.4	39.5
GridConv	-	-	39.0
D-GridConv	-	-	**37.1**

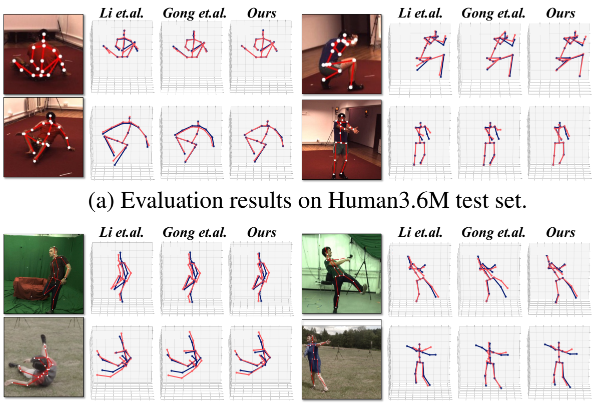
Table 4: Ablation study of applying grid pose on GCN methods. We report MPJPE under ground truth input.

Ablation Study

Finally, we provide a lot of ablative experiments to study different components and design aspects of our GLN. All ablative experiments are performed on Human3.6M dataset.

Grid versus Graph. Grid pose has a similar structure to graph pose, which makes it possible to be combined in graph convolution framework. We investigate the combination by setting two kinds of input pose: (1) graph-structured pose having handcrafted-grid topology, denoted as G_{craft}; (2) handcrafted grid pose, denoted as D_{craft}. We select two GCN baselines LCN and SemGCN. For D_{craft}, we modify their convolution layers to receive 5×5 grid input. Results shown in Table 4 demonstrate three facts. First, results on G_{craft} (43.8→41.8, 39.7→39.4) indicate that grid-based topology is helpful to pose learning. Second, results on D_{craft} indicate that employing grid pose on GCN does not ensure better performance (41.8→43.5, 39.4→39.5). Last, GridConv family performing better indicates that grid convolution is more effective and more suitable for grid pose.

SGT designs for constructing grid pose. Although we provide both handcrafted and learnable SGT designs, a more straightforward SGT design is to generate a grid pose randomly. Hence it is necessary to compare their effectiveness. Accordingly, we conducted a set of ablative experiments using these three designs separately to construct 5×5 grid pose,



(a) Evaluation results on Human3.6M test set.

(b) Cross-dataset results on MPI-INF-3DHP test set.

Figure 5: Visualization comparison between top-performing methods and GLN. GT in blue and prediction in red.

Grid Pose by	P1	P1*	P2
Random SGT #1	<u>49.3</u>	38.5	38.5
Random SGT #2	49.5	38.2	38.5
Random SGT #3	49.5	37.9	<u>38.3</u>
Random SGT #4	49.6	<u>37.8</u>	38.4
Mean over #1-4	49.5	38.1	38.4
Handcrafted SGT	47.9$_{2.8\%\downarrow}$	37.1$_{1.9\%\downarrow}$	37.9$_{1.0\%\downarrow}$
Learnable SGT	47.6$_{0.6\%\downarrow}$	36.4$_{1.9\%\downarrow}$	37.4$_{1.3\%\downarrow}$

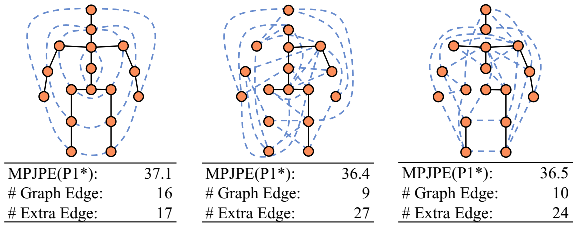
Table 5: Ablation study on GLN using different SGT designs to construct grid pose. The size of grid pose is fixed to 5×5.

and report results in Table 5. When generating a random grid pose, each joint is forced to appear at least once. Surprisingly, it can be observed that random grid layouts show good performance even with no semantic skeleton topology constraint contained. Comparatively, our handcrafted and learnable designs are obviously better than random ones.

Analysis of learnt grid pose patterns. To have a better understanding of learnable SGT design, in Figure 6, we visualize two learnt grid pose patterns converted into an equivalent graph structure where dotted edges denote neighboring joints in grid pose. Tthe learnt grid pose patterns maintain fewer skeleton edges, yet establish new edges to keep all graph nodes reachable, which include many long-distance connections (e.g. head to knees).

Grid size. A critical question is how to select a proper size $H{\times}P$ for grid pose. Accordingly, we conducted an experiment to compare the performance of training our lifting network with different $H{\times}P$ settings of grid pose. Detailed results are shown in Figure 7. It demonstrates that 5×5 size reaches the best performance, hence is set as the default.

Grid pose with different body joint numbers. A skeleton pose with 17 body joints is used in our main experiments, following many existing works. Table 6 shows results of investigating the generalization ability of our method with another skeleton of 32 body joints. Additional joints enrich motion information and make the task more challenging.



(a) Handcrafted. (b) Learnt #1. (c) Learnt #2.

Figure 6: Visualization of handcrafted and learnt grid pose patterns converted into an equivalent graph structure.

Body Joint # (J) Grid Pose Size ($H{\times}P$)	17 5×5	32 7×5	Δ
Handcrafted SGT	37.1	**35.0**	2.1
Learnable SGT	36.4	**33.4**	3.0

Table 6: Ablation study on using different numbers of body joints as input. We report MPJPE under ground truth input.

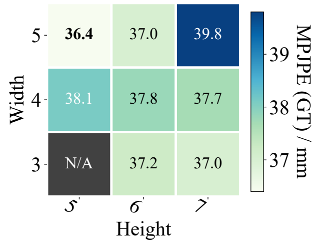


Figure 7: Ablation study on the effect of grid size $H{\times}P$.

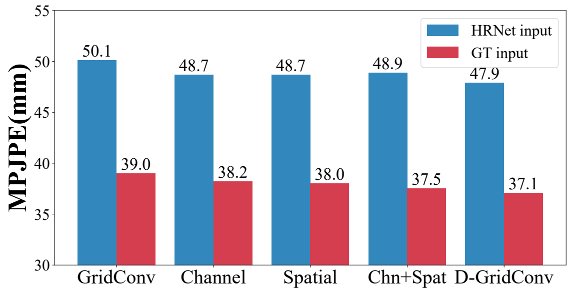


Figure 8: Ablation study on different attention designs.

Attention designs. Regarding the attention module for D-GridConv, we also tried several other designs besides the proposed one including: (a) channel-wise attention, similar to SENet (Hu, Shen, and Sun 2018); (b) spatial-wise attention, a reduced version of CBAM (Woo et al. 2018); (c) spatial+channel attention, similar to CBAM. Figure 8 compares the performance, showing our design performs the best.

Conclusion

We take the lead in extending convolution operations to estimate 3D human pose from 2D detection by shifting pose representation from graph to weave-like grid pose through Semantic Grid Transformation. Based on grid layout, we formulate grid convolution and construct grid lifting network. Extensive experiments on two public benchmarks demonstrate superiority of our method to previous works.

Acknowledgements

The authors would like to acknowledge the support from NSFC (No. 62072449, No. 61972271) and Macau Sci and Tech Fund (0018/2019/AKP).

References

Andriluka, M.; Roth, S.; and Schiele, B. 2010. Monocular 3d pose estimation and tracking by detection. In *Proceeding of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.

Belagiannis, V.; Amin, S.; Andriluka, M.; Schiele, B.; Navab, N.; and Ilic, S. 2014. 3D pictorial structures for multiple human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Cai, Y.; Ge, L.; Liu, J.; Cai, J.; Cham, T.-J.; Yuan, J.; and Thalmann, N. M. 2019. Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Cao, Z.; Hidalgo, G.; Simon, T.; Wei, S.-E.; and Sheikh, Y. 2019. OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(1).

Ci, H.; Wang, C.; Ma, X.; and Wang, Y. 2019. Optimizing network structure for 3d human pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Courbariaux, M.; Bengio, Y.; and David, J.-P. 2015. Binaryconnect: Training deep neural networks with binary weights during propagations. *Proceedings of the Advances in Neural Information Processing Systems*, 28.

Dabral, R.; Mundhada, A.; Kusupati, U.; Afaque, S.; Sharma, A.; and Jain, A. 2018. Learning 3D Human Pose from Structure and Motion. In *Proceedings of the European Conference on Computer Vision*.

Fang, H.-S.; Xu, Y.; Wang, W.; Liu, X.; and Zhu, S.-C. 2018. Learning pose grammar to encode human body configuration for 3d pose estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Gong, K.; Zhang, J.; and Feng, J. 2021. PoseAug: A Differentiable Pose Augmentation Framework for 3D Human Pose Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Hu, J.; Shen, L.; and Sun, G. 2018. Squeeze-and-excitation networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Hu, W.; Zhang, C.; Zhan, F.; Zhang, L.; and Wong, T.-T. 2021. Conditional directed graph convolution for 3d human pose estimation. In *Proceedings of the 29th ACM International Conference on Multimedia*.

Ionescu, C.; Papava, D.; Olaru, V.; and Sminchisescu, C. 2014. 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).

Jang, E.; Gu, S.; and Poole, B. 2017. Categorical Reparameterization with Gumbel-Softmax. *Proceedings of the International Conference on Learning Representations*.

Kanazawa, A.; Black, M. J.; Jacobs, D. W.; and Malik, J. 2018. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Kang, Y.; Yao, A.; Wang, S.; Lu, M.; Chen, Y.; and Wu, E. 2020. Explicit Residual Descent for 3D Human Pose Estimation from 2D Joint Locations. In *Proceedings of the British Machine Vision Conference*.

Lee, K.; Lee, I.; and Lee, S. 2018. Propagating lstm: 3d pose estimation based on joint interdependency. In *Proceedings of the European Conference on Computer Vision*.

Li, C.; and Lee, G. H. 2019. Generating multiple hypotheses for 3d human pose estimation with mixture density network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Li, S.; Ke, L.; Pratama, K.; Tai, Y.-W.; Tang, C.-K.; and Cheng, K.-T. 2020. Cascaded deep monocular 3D human pose estimation with evolutionary training data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Liu, K.; Ding, R.; Zou, Z.; Wang, L.; and Tang, W. 2020. A comprehensive study of weight sharing in graph networks for 3d human pose estimation. In *Proceedings of the European Conference on Computer Vision*.

Luo, C.; Chu, X.; and Yuille, A. 2018. OriNet: A Fully Convolutional Network for 3D Human Pose Estimation. In *Proceedings of the British Machine Vision Conference*.

Martinez, J.; Hossain, R.; Romero, J.; and Little, J. J. 2017. A simple yet effective baseline for 3d human pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Mehta, D.; Rhodin, H.; Casas, D.; Fua, P.; Sotnychenko, O.; Xu, W.; and Theobalt, C. 2017a. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *Proceedings of International Conference on 3D Vision*.

Mehta, D.; Sotnychenko, O.; Mueller, F.; Xu, W.; Sridhar, S.; Pons-Moll, G.; and Theobalt, C. 2018. Single-shot multi-person 3d pose estimation from monocular rgb. In *Proceedings of International Conference on 3D Vision*.

Mehta, D.; Sridhar, S.; Sotnychenko, O.; Rhodin, H.; Shafiei, M.; Seidel, H.-P.; Xu, W.; Casas, D.; and Theobalt, C. 2017b. Vnect: Real-time 3d human pose estimation with a single rgb camera. *ACM Transactions on Graphics*, 36(4).

Moon, G.; Chang, J. Y.; and Lee, K. M. 2019. Camera distance-aware top-down approach for 3d multi-person pose estimation from a single rgb image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Moon, G.; and Lee, K. M. 2020. I2l-meshnet: Image-to-lixel prediction network for accurate 3d human pose and mesh estimation from a single rgb image. In *Proceedings of the European Conference on Computer Vision*.

Pavlakos, G.; Zhou, X.; Derpanis, K. G.; and Daniilidis, K. 2017. Coarse-to-fine volumetric prediction for single-image

3D human pose. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Pavllo, D.; Feichtenhofer, C.; Grangier, D.; and Auli, M. 2019. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Rogez, G.; Weinzaepfel, P.; and Schmid, C. 2017. Lcr-net: Localization-classification-regression for human pose. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Sharma, S.; Varigonda, P. T.; Bindal, P.; Sharma, A.; and Jain, A. 2019. Monocular 3d human pose estimation by generation and ordinal ranking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Sun, K.; Xiao, B.; Liu, D.; and Wang, J. 2019. Deep High-Resolution Representation Learning for Human Pose Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Sun, X.; Shang, J.; Liang, S.; and Wei, Y. 2017. Compositional human pose regression. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Sun, X.; Xiao, B.; Wei, F.; Liang, S.; and Wei, Y. 2018. Integral human pose regression. In *Proceedings of the European Conference on Computer Vision*.

Woo, S.; Park, J.; Lee, J.-Y.; and Kweon, I. S. 2018. Cbam: Convolutional block attention module. In *Proceedings of the European Conference on Computer Vision*.

Yang, W.; Ouyang, W.; Wang, X.; Ren, J.; Li, H.; and Wang, X. 2018. 3d human pose estimation in the wild by adversarial learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Yu, F.; Salzmann, M.; Fua, P.; and Rhodin, H. 2021. PCLs: Geometry-aware neural reconstruction of 3D pose with perspective crop layers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Zeng, A.; Sun, X.; Huang, F.; Liu, M.; Xu, Q.; and Lin, S. 2020. Srnet: Improving generalization in 3d human pose estimation with a split-and-recombine approach. In *Proceedings of the European Conference on Computer Vision*.

Zeng, A.; Sun, X.; Yang, L.; Zhao, N.; Liu, M.; and Xu, Q. 2021. Learning skeletal graph neural networks for hard 3d pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Zhao, L.; Peng, X.; Tian, Y.; Kapadia, M.; and Metaxas, D. N. 2019. Semantic graph convolutional networks for 3d human pose regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Zhou, K.; Han, X.; Jiang, N.; Jia, K.; and Lu, J. 2021. HEMlets PoSh: Learning Part-Centric Heatmap Triplets for 3D Human Pose and Shape Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zhou, X.; Huang, Q.; Sun, X.; Xue, X.; and Wei, Y. 2017. Towards 3d human pose estimation in the wild: a weakly-supervised approach. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Zhou, X.; Zhu, M.; Leonardos, S.; Derpanis, K. G.; and Daniilidis, K. 2016. Sparseness meets deepness: 3d human pose estimation from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Zou, Z.; Liu, K.; 0003, L. W.; and Tang, W. 2020. High-order Graph Convolutional Networks for 3D Human Pose Estimation. In *Proceedings of the British Machine Vision Conference*.