Kinematics Modeling Network for Video-based Human Pose Estimation

Yonghao Dang[a], Jianqin Yin[a,*], Shaojie Zhang[a], Jiping Liu[b,c,d], Yanzhu Hu[e]

[a]*School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing, China*
[b]*Chinese Academy of Surveying and Mapping, Beijing, China*
[c]*Faculty of Geosciences and Environmental Engineering, Southwest Jiaotong University, Chengdu, China*
[d]*School of Resource and Environmental Sciences, Wuhan University, Wuhan, China*
[e]*School of Modern Post (School of Automation), Beijing University of Posts and Telecommunications, Beijing, China*

Abstract

Estimating human poses from videos is critical in human-computer interaction. Joints cooperate rather than move independently during human movement. There are both spatial and temporal correlations between joints. Despite the positive results of previous approaches, most of them focus on modeling the spatial correlation between joints while only straightforwardly integrating features along the temporal dimension, which ignores the temporal correlation between joints. In this work, we propose a plug-and-play kinematics modeling module (KMM) to explicitly model temporal correlations between joints across different frames by calculating their temporal similarity. In this way, KMM can capture motion cues of the current joint relative to all joints in different time. Besides, we formulate video-based human pose estimation as a Markov Decision Process and design a novel kinematics modeling network (KIMNet) to simulate the Markov Chain, allowing KIMNet to locate joints recursively. Our approach achieves state-of-the-art results on two challenging benchmarks. In particular, KIMNet shows robustness to the occlusion. Code will be released at https://github.com/YHDang/KIMNet.

*Corresponding author
Email address: dyh2018@bupt.edu.cn (Yonghao Dang)

Keywords: human pose estimation, relational modeling, keypoint detection

1. Introduction

Human pose estimation (HPE) plays a fundamental role in pattern recognition. Pose sequences exhibit resilience against superficial visual variations, such as the background, clothing texture, and illumination conditions [1], enabling pose sequences to offer pure action representations for other computer vision tasks, including action recognition [2], person re-identification [3], and human parsing [4]. Enhancing the performance of HPE is significant for advancing the field of pattern recognition. According to data type, HPE can be roughly divided into image-based and video-based HPE. Unlike images, videos contain appearance features for each frame and encapsulate valuable temporal information. As a result, effectively modeling this temporal information becomes crucial for accurately estimating poses from videos.

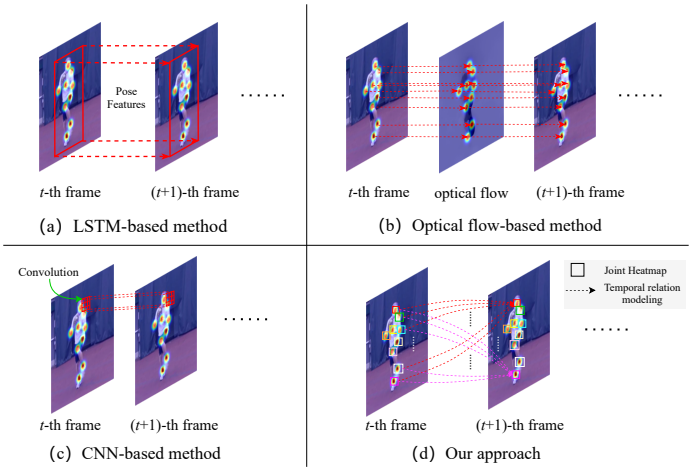


Figure 1: Comparisons between our approach and other existing methods. (a) LSTM-based methods. (b) Optical flow-based approaches. (c) CNN-based methods. (d) The proposed KIMNet.

Due to its remarkable progress in the computer vision community, deep learning approaches have been commonly used to model temporal information

for video-based HPE. As shown in Figure1, LSTM-based methods [5, 6] aggregate vanilla sequential features of poses in adjacent frames. Optical flow-based [7, 8] approaches align the same joints in different frames through the joints' optical flow information. CNN-based [9, 10, 11, 12] methods model the motion information of poses by integrating local features within the convolutional kernel along the temporal dimension. Despite the positive results shown in previous methods, there is a commonly essential but neglected factor among these approaches. Most methods model the motion information of the poses in the temporal dimension but ignore the temporal correlation between different joints.

The joints of the human associate with each other rather than moving independently during human movement. In other words, there is simultaneous cooperation among all joints to maintain coordination of actions [13]. The cooperation of joints can be viewed as the spatial and temporal dependency between joints. The spatial dependency between joints can be regarded as the pose structural information that is represented by the spatial correlations between joints [12]. Meanwhile, the temporal dependency between joints can be seen as the relative motion between different joints. For a clip of video, the joints in frame $t+1$ move relative to all joints in frame t. This relative motion can be expressed as the temporal correlation between joints to some content. Based on the temporal correlation, the model can infer the positions of joints in the current frame via the information about joints in the previous frames. The model can infer its position for the occluded joint according to the information of joints related to the occluded joint in other frames. Therefore, the temporal dependency between joints is significant for locating joints. However, most approaches pay much attention to modeling the pose structural information, while ignoring the temporal correlation between joints, which limits the model's performance for estimating poses from videos.

In this paper, we propose a **KI**nematics **M**odeling based **Net**work (KIMNet) to locate joints by explicitly modeling the temporal dependency between joints across different frames. Specifically, we present a novel and plug-and-play kinematic modeling module (KMM) based on the attention mechanism to explicitly

explore the temporal correlation between joints by modeling their temporal similarity. In this way, the KMM learns the motion cue of each joint relative to all joints in the previous frame. By combining motion cues and the historically positional information of poses, KMM can preliminarily predict the initial positions of joints from the current frame in advance. Based on this, we formulate the video-based human pose estimation as a Markov decision process. The proposed KIMNet recursively estimates human poses from each frame. Besides, thanks to the temporal dependency, the proposed KIMNet locates the current joint by aggregating the information of other joints related to the current joint instead of locating it only based on its own features.

Contributions of this work are summarized as follows.

- We propose a plug-and-play kinematics modeling module (KMM) based on the attention mechanism to explicitly model the temporal correlation between joints across different frames. The KMM can predict the initial joints' positions in advance by aggregating joints' motion information and historical positions.

- We formulate video-based human pose estimation as a Markov decision process and present a **KI**nematics **M**odeling based **Net**work (KIMNet) to simulate it. With the guidance of the temporal correlation, KIMNet locates the current joint by integrating the information of other joints from other frames rather than only depending on the current joint's information, which improves the model's robustness against the occlusion.

- The proposed KIMNet achieves new state-of-the-art performance among methods on the challenging Penn Action and Sub-JHMDB datasets. Furthermore, Experimental results demonstrate that the KMM is compatible with existing pose estimation frameworks.

2. Related Works

2.1. Video-based Human Pose Estimation

Human pose estimation has always been one of the hot issues in computer vision tasks. With the improvement of deep learning, image-based pose estimation [14, 15, 16, 17] has made significant progress. However, due to the lack of temporal modeling, image-based HPE approaches are difficult to maintain superior performance in video-based HPE.

There has been a lot of work trying to explore temporal information of poses. The optical flow [7, 8] was first used to align joints across different frames. These methods have achieved attractive results while suffering from high computational cost. [5, 6] applied LSTM to extract temporal features. LSTM shows superior performance on temporal modeling, but the above methods are constrained to model the sequential correlation based on image features simply. Recently, CNN [9, 11, 18] is commonly used to extract spatio-temporal representation of poses. The spatio-temporal features are usually limited by the receptive field of convolutional kernels for these CNN-based approaches. [19] proposed a diffusion architecture to aggregate visual evidence across frames. [20] leveraged a hierarchical framework to capture coarse-to-fine deformations of poses across frames. Unlike these methods that focus on enhancing the temporal features of poses, the proposed method aims to capture the motion of poses by explicitly modeling the temporal correlation between joints across frames.

2.2. Relation Modeling in Human Pose Estimation

The effectiveness of relation modeling has been demonstrated in many fields, such as image re-ranking [21], image [22] and action recognition [1]. Yu et al. [21] proposed a multimodal hypergraph learning-based sparse coding to explore the complementarity of different features. Zhang et al.[22] presented a vector of locally and adaptively aggregated descriptors to improve image feature representation. Compared with the image, the human body is more structural because of the apparent connection between joints. To model the

structural information of the human body, Bin et al. [17] utilized a graphic convolution to model the physical connection between joints. [23] designed a graph structure network (GSN) to locate invisible joints by aggregating other joints' information. Jiang et al. [24] pointed out that multi-scale features of poses are significant for locating joints and proposed a pyramid gating network to capture semantic features of poses. Yang et al.[25] used pairwise attention to explore the association between joints. Dang et al. [12] introduced a relation modeling module based on an attention mechanism to explore the structural information of poses. Yang et al. [25] used a self-attention mechanism to explore the relationship between different instances' joints. Similarly, [26, 27] used a transformer to model the spatial correlations between pose tokens. Most methods mentioned above focus on spatial correlation modeling. Gai et al. [28] proposed an SLT-Pose to strengthen the interaction of poses between the target frame and the local sequence through the cross-attention mechanism. Different from [28] that explore the pose-level interaction across frames, the proposed KIMNet is able to model the joint-level temporal correlations.

2.3. Temporal Correlation Modeling in Video Analysis

Temporal correlations have been widely used in various tasks. Wu et al. [29] used the attention mechanism for integrating spatiotemporal features in a person re-identification task. Wang et al. introduced pixel-wise contrastive algorithm [30] and co-attention classifier [31] to associate the salient objects for the semantic segmentation task. Furthermore, Wang et al. proposed COSNet [32] with the group co-attention mechanism [33] to predict object masks and associate them across multiple frames for the video-based object segmentation task. Similar to video-based object segmentation, object tracking also needs to match the same target in different frames. Wu et al. [34] presented an online TraDeS tracker that assigns the same target in multiple frames by calculating a pixel-wise similarity between adjacent frames. The above approaches usually aim at associating objects from different frames by modeling their correlations at the pixel level. For the human pose estimation, it is worth focusing on the

correlation between the whole joints (represented by feature maps or heatmaps). The pixel-level correlation between joints' features will destroy the integrity of joints to some content. To solve this issue, KMM is proposed to model the temporal correlations between joints across frames.

3. Methodology

3.1. Problem Statement

We formulate the video-based human pose estimation as a Markov decision process in this paper. There are slight changes between poses in adjacent frames. When the pose in frame t is given, we can roughly infer the corresponding pose in the next frame. Based on the above observation, the temporal correlation of the pose can be simplified into a Markov decision process [35], that is, the pose at the current moment is only related to the previous moment. Given the state at time t (*i.e.*, the human pose m_t), the state at time $t+1$ (m_{t+1}) can be denoted as follows.

$$P\left(m_{t+1}|m_t, m_{t-1}, \cdots, m_1\right) = P\left(m_{t+1}|m_t\right) \tag{1}$$

where $P(\cdot)$ is the state transition matrix. Eq. 1 shows that the state at time $t+1$ can be obtained based on the state at time t.

For a series of sequential frames (*i.e.*, $\mathcal{I} = \{I_t, I_{t+1}, \cdots, I_{t+T}\}$) randomly sampled from a video, we take joints' heatmaps, $M = \{m_t, m_{t+1}, \cdots, m_{t+T}\}$, as states in T continuous frames. We can obtain the observation $F(I_t)$ from frame t via a function $F(\cdot)$. Based on the observation values of two adjacent frames, the target's latent motion ($\mathcal{O}_{t,t+1}$) can be modelled through a motion function $\phi(\cdot)$. Combining the motion information and the state at frame t will preliminarily predict the state for the frame $t+1$. The predicted state is uncertain due to the lack of information about frame $t+1$. Therefore, the observation of frame $t+1$ (F_{t+1}) is used to balance the above uncertainty. And the accurate state of frame

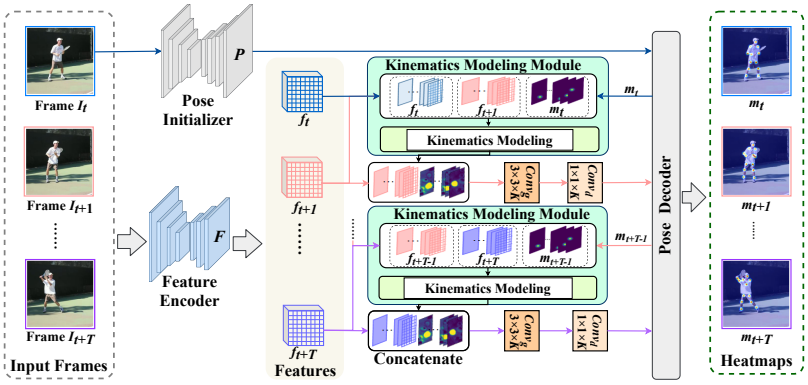


Figure 2: Overview of the proposed kinematics modeling network.

$t+1$ (m_{t+1}) can be formulated as follows.

$$\begin{aligned} m_{t+1} &= \psi(\zeta\left(\mathcal{O}_{t,t+1}, m_t\right) + F(I_{t+1})) \\ \mathcal{O}_{t,t+1} &= \phi(F(I_t), F(I_{t+1})) \end{aligned} \tag{2}$$

where $\zeta(\cdot)$ represents the initial state prediction function. $\psi(\cdot)$ is the correction function to generate the correct state for frame $t+1$. $\phi(\cdot)$ is the motion function used to model the dynamics of the target in continuous frames. In practice, we take joints' heatmaps and appearance features as the states and observations. Function $\psi(\cdot)$, $\phi(\cdot)$, and $F(\cdot)$ are implemented by the convolutional blocks. Details are described in *Subsection 3.2*.

3.2. KIMNet Model for Video-based Pose Estimation

We design a kinematics modeling network (KIMNet) to fit equation 2, as shown in Figure 2. As stated in eq. 2, there are four critical steps in KIMNet's modeling process, including acquisition of observations and initial state, motion modeling of joints, prediction of the state, and correction of the state.

Acquisition of observations and initial state. We use a feature encoder (*i.e.*, F) based on the convolutional neural network to extract appearance features from each input frame as the observation (*i.e.*, f_t). Moreover, we adopt a pretrained pose initializer to estimate the initial pose from the first frame as the

initial state. For the convenience of description, we take the frame $(t+1)$ as the current frame and the frame t as the historical frame in this paper.

Motion modeling of joints. Human motion cues are significant for estimating poses from videos. If the motion cues are modeled effectively, it is possible to infer joints' positions in frame $t+1$ by combining the motion cues with the historical positions of joints. To extract robust motion cues, we propose a kinematic modeling module (KMM) to explore each joint's dynamical information by modeling the temporal correlation between any two joints. Then each joint's motion cues are combined with its historical position in frame t to predict its positions in frame $t+1$.

As mentioned above, the motion information is modeled by modeling the temporal similarity between joint features f_t and f_{t+1}. Joints are represented by a set of feature maps consisting of $h \times w$ pixels during modeling process. The motion of the whole feature map, that is, the motion of all pixels should be modeled to represent the motion of the joint. Considering that the dot-product can measure the correlation [12]. Furthermore, the dot-product is essentially a weighted sum that integrates the information of all pixels in the feature map. Thus, it can reflect the motion of the whole joint's feature map to some extent.

$$O_{t,t+1} = \phi\left(f_t, f_{t+1}\right), t = 1, 2, ..., T-1 \tag{3}$$

where $\phi(\cdot)$ is the motion function that is implemented based on the dot-product to model the temporal dependency between joints across frames by calculating the temporal similarity between any them. $O_{t,t+1} \in \mathbb{R}^{K \times K}$ reflects the movement of the joint in frame $t+1$ relative to all joints in frame t.

Prediction of the state. $O_{t,t+1}$ contains the motion information about each joint, and m_t includes the positional information of joints in the t-th frame. By fusing the positional information and motion cues of joints, the model can preliminarily predict the joints' positions in frame $t+1$ as follows.

$$m_{t+1}^{p} = \zeta(O_{t,t+1}, m_t), t = 1, 2, \ldots, T-1 \tag{4}$$

where $m_{t+1}^{p} \in \mathbb{R}^{h \times w \times K}$ is the initial state, *i.e.*, predicted joint heatmaps in

frame $t+1$. $\zeta(\cdot)$ is the prediction function based on the dot-product. Since each element in $O_{t,t+1}$ reflects the motion degree of j-th joint J_{t+1}^{j} in frame $t+1$ related to J_{t}^{i} in frame t, the model can predict the joints' initial position m_{t+1}^{p} by conducting the weighted sum between motion information $O_{t,t+1}$ and position information m_t.

Correction of the state. To obtain precise joint heatmaps, we apply a pose decoder to produce the final heatmaps. For the first frame, the pose decoder is directly applied to the initial pose predicted by the pose initializer to get the final pose.

For subsequent frames, there are some uncertainties in predicted states m_{t+1}^{p} due to the lack of information for frame $t+1$. Therefore, we introduce the observation from frame $t+1$ (*i.e.*, joint features f_{t+1}) into the current predicted state to alleviate these uncertainties. Practically, we adopt a correction function $\psi(\cdot)$ consisting of a 3×3 and a 1×1 convolutions (*i.e.*, $Conv_g$ and $Conv_d$) to fuse the pose features and predicted poses.

$$\begin{aligned} f_{t+1}^{coarse} &= \sum_{i=1}^{K} w_g^i * m_{t+1}^{p} + \sum_{j=K+1}^{D} w_g^j * f_{t+1} \\ f_{t+1}^{fine} &= \sum_{i=1}^{K} w_d^i * f_{t+1}^{coarse} \end{aligned} \tag{5}$$

where $f_{t+1}^{coarse} \in \mathbb{R}^{h\times w\times D}$ and $f_{t+1}^{fine} \in \mathbb{R}^{h\times w\times K}$ represent the coarse- and fine-grained pose representations, respectively. $w_g^i \in \mathbb{R}^{3\times3}$ and $w_d^i \in \mathbb{R}^{1\times1}$ are convolutional filters in $Conv_g$ and $Conv_d$. And $*$ represents the convolutional operation. In practice, KIMNet takes the concatenation both of m_{t+1}^{p} and f_{t+1} as the input, and conducts $Conv_g$ to generate the coarse pose feature, which is equivalent to the sum of performing $Conv_g$ on m_{t+1}^{p} and then on f_{t+1}. After $Conv_g$ and $Conv_d$, we obtain the coarse-to-fine pose features.

Finally, the pose decoder is used to produce the final heatmap for each joint.

$$m_{t+1} = \text{Decoder}\left(f_{t+1}^{fine}\right), t = 1, 2, ..., T-1 \tag{6}$$

where Decoder$(\cdot)$ is implemented by the joint relation extractor [12].

3.3. Kinematics Modeling Module

In order to model the temporal correlations between joints, we present a plug-and-play kinematics modeling module (KMM) based on the attention mechanism, as shown in Figure 3. KMM fits the motion function $\phi(\cdot)$ and the initial state prediction function $\zeta(\cdot)$ in eq. 2, where the motion function $\phi(\cdot)$ contains two steps: robust pose representation extraction and temporal dependency modeling.

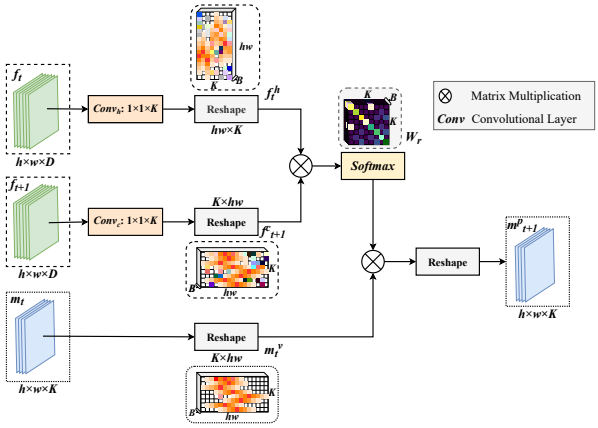


Figure 3: The structure of the proposed kinematics modeling module.

Robust pose representations extraction. The information of the adjacent two frames is similar, resulting in the similarity between appearance feature f_t and f_{t+1}. To explore robust motion information of poses, we adopt two independent 1×1 convolutions denoted as $Conv_h$ and $Conv_c$ to extract discriminative pose representations from f_t and f_{t+1}. Then reshaping operation is used to convert the feature matrix into a feature vector for facilitating the temporal correlation modeling.

$$\begin{aligned} f_t^h &= \text{Reshape}\left(\sigma(W_h * f_t)\right) \in \mathbb{R}^{(h\times w)\times K} \\ f_{t+1}^c &= \text{Reshape}\left(\sigma(W_c * f_{t+1})\right) \in \mathbb{R}^{K\times(h\times w)} \end{aligned} \tag{7}$$

where $f_t \in \mathbb{R}^{h\times w\times D}$ and $f_{t+1} \in \mathbb{R}^{h\times w\times D}$ are joint features in frame t and $t+1$. $W_h \in \mathbb{R}^{1\times 1\times K}$ and $W_c \in \mathbb{R}^{1\times 1\times K}$ are weights of $Conv_h$ and $Conv_c$. $\sigma(\cdot)$ and

$*$ represents the nonlinear function and convolutional operation, respectively. Each column in f_t^h and each row f_{t+1}^c represent one joint.

Temporal dependency modeling. Considering that joints are relevant to each other when the person is moving, we model the temporal correlation between any two joints by calculating their temporal similarity. Specifically, the dot-product can measure the similarity between two vectors to some extent. We apply the dot-product of pose features f_t^h and f_{t+1}^c to model the temporal correlation between any two joints across different frames. For the i-th and j-th joints represented by $f_t^{h_i} \in \mathbb{R}^{(h\times w)\times 1}$ and $f_{t+1}^{c_j} \in \mathbb{R}^{1\times(h\times w)}$, the temporal correlation between them can be modeled as follows.

$$o_{t,t+1}^{i,j} = f_{t+1}^{c_j} \cdot f_t^{h_i}, t = 1, 2, \ldots, T-1 \tag{8}$$

where $o_{t,t+1}^{i,j}$ represents the temporal dependency between the joint J_t^i and the joint J_{t+1}^j. Because the dot-product is essentially a weighted sum of joints' feature maps, it reflects the overall motion of the joint J_{t+1}^j relative to J_t^i. Similarly, the temporal dependency between J_{t+1}^j and other joints in frame t can also be captured following Eq.8. This process can be realized by the matrix multiplication as follows.

$$O_{t,t+1} = \left(\frac{f_{t+1}^c \otimes f_t^h}{\sqrt{d}}\right) \in \mathbb{R}^{K\times K}, t = 1, 2, \ldots, T-1 \tag{9}$$

where $\otimes$ denotes matrix multiplication used to model the temporal correlation between any two joints in different frames. $O_{t,t+1} = \{o_{t,t+1}^{1,1}, o_{t,t+1}^{1,2}, \cdots, o_{t,t+1}^{i,j}, \cdots, o_{t,t+1}^{K,K}\} \in \mathbb{R}^{K\times K}$ represents the temporal dependency between joints across frames t and $t+1$. d is the normalized factor that is equal to the dimension of features f_t^h and f_{t+1}^c.

In order to enhance the motion cues of the joint J_{t+1}^j, the softmax operation, then, is used to model the global correlation [36] that can be regarded as the dependency between the J_{t+1}^j and all joints in the t-th frame.

$$w_r^{i,j} = \frac{1}{\sum_{l=1}^{K} o_{t,t+1}^{l,j}} \cdot o_{t,t+1}^{i,j}, \quad \begin{matrix} t = 1, 2, \ldots, T-1 \\ i = 1, 2, \ldots, K \end{matrix} \tag{10}$$

where $w_r^{i,j}$ is the temporal attention weight between J_{t+1}^j and J_t^i, reflecting the relative motion degree between two joints. $W_r = \{w_r^{1,1}, w_r^{1,2}, \cdots, w_r^{K,K}\} \in \mathbb{R}^{K\times K}$ highlights the attention of joints that are closely related to the current moving joint. Because $o_{t,t+1}^{i,j}$ includes the temporal dependency between two joints across different frames, the sum of $o_{t,t+1}^{l,j}$ integrates temporal dependency of all joint pairs. Therefore, the Eq.10 can reflect the motion information of J_{t+1}^j relative to all joints in the t-th frame to some extent.

Prediction of joints' initial positions. By fusing the known state (*i.e.*, the historical joint's heatmap m_t) and the motion information, intuitively, the model is able to predict the initial state for frame $t+1$ (*i.e.*, m_{t+1}^p) in advance. Since W_r includes the motion information of joints, we apply W_r to joint's position in frame t to predict the k-th joint's position in frame $t+1$. The KMM computes the response at a position of J_{t+1}^k as a weighted sum of the positions of all joints at frame t, which can be denoted as:

$$\bar{m}_{t+1}^p = W_r \cdot m_t^v \in \mathbb{R}^{K\times(h\times w)} \tag{11}$$

where $\bar{m}_{t+1}^p$ is the initial heatmap vector of joints. In practice, we transfer the heatmap m_t to ${m_t}^v \in \mathbb{R}^{K\times(h\times w)}$ for the convenient inference of m_{t+1}^p. The reshaping operation is used to convert $\bar{m}_{t+1}^p$ to the initial heatmap m_{t+1}^p with the shape of $K \times h \times w$.

3.4. Training Loss Function

In this paper, we minimize the L_2 norm between the output of the model and the heatmap of ground truth to optimize the proposed KIMNet. Specifically,

$$Loss = \frac{1}{T}\sum_{t=1}^{T} \|M_t - M_t'\|_2^2, t = 1, 2, \ldots, T \tag{12}$$

where $M_t = \{{m_t}^1, {m_t}^2, \ldots, {m_t}^K\}$ denotes the output of KIMNet. $M_t' = \{{m_t'}^1, {m_t'}^2, \ldots, {m_t'}^K\}$ is the joints' heatmap generated according to the ground truth. T is the total number of training frames. $\|\cdot\|$ represents the L_2 norm.

3.5. Comparisons with RPSTN

Although the structure of the proposed KIMNet is similar to [12], there are essential differences between them. **1. Motivations are different.** [12] aims to explore the spatial affinity among joints by modeling the spatial correlations between joints within a frame. While the proposed KIMNet is designed to model the temporal dependency among joints across different frames. **2. Temporal modeling approaches are different.** [12] uses several convolutional layers to transfer the historical pose knowledge and template matching to search for similar regions in the current frame, which ignores the temporal correlation between joints in different frames. In addition, as the number of convolution layers increases, the feature resolution decreases continuously, leading to the loss of historical pose information to a certain extent. In contrast to [12], the proposed KIMNet explicitly models the temporal correlation between joints across different frames. In this way, the model can locate the current joint by aggregating joints' information in the previous frame. Furthermore, the proposed KMM is implemented via attention mechanism, which preserves the original resolution of pose features while propagating historical features. Compared with RPSTN [12], KIMNet can avoid the loss of pose information due to the reduction of the resolution.

4. Experiments

We first introduce specific experimental settings. Then, we compare the proposed KIMNet against existing state-of-the-art video-based methods. We also provide ablation studies to confirm the effectiveness of the proposed KMM. Finally, we conduct comprehensive experiments in the occluded scene to show advantages of the proposed model.

4.1. Experimental Settings

4.1.1. Datasets

Penn Action dataset contains 2326 video clips. Each frame is annotated with 13 joints, including the head, shoulders, elbows, wrists, hips, knees, and

ankles. Besides, the Penn Action dataset annotates the position of the person, *i.e.*, the bounding box, and gives visibility for each joint. Keeping the consistency with [11, 12], we train the model with 1258 video clips, and the others are used to evaluate the model.

Sub-JHMDB dataset includes 316 videos. There are 15 body joints in each frame. Following [12], 3-fold-cross-validation is used to evaluate the model. According to existing methods, training sets include 227, 236, and 224 video clips, respectively, and testing sets correspondingly contain 89, 80, and 92 samples. To make a fair comparison with previous approaches, we take the average result on three testing sets as the final result.

4.1.2. Implementation details

Following [12], SimpleBaseline [15] takes ResNet-101 as the backbone and is used as the pose initializer. SimpleBaseline [15] that takes ResNet-50 as the backbone is chosen as the feature encoder. Both pose initializer and feature encoder are pre-trained on the MPII [37] dataset. Furthermore, to make a fair comparison with existing methods, keeping consistency with works [11, 12], we also: randomly select 5 contiguous frames from each training video clip; adopt the same data augmentation settings as [11, 12] including random scaling, rotation, and flipping; use the Adam [38] optimizer to train the model and the training epoch is set to 100 following [12]. The learning rate is initialized to 0.005, and the batch size is set to 16 on the Penn Action dataset. Because the scale of the Sub-JHMDB dataset is smaller than that of Penn Action dataset, the learning rate and batch size are set to 0.001 and 8. All experiments are conducted on two NVIDIA GeForce RTX 3080Ti GPUs.

4.1.3. Evaluation metric

The percentage of correct keypoints (PCK) is used as the evaluation metric. The PCK for each joint is calculated as follows.

$$PCK_k = \frac{\sum_{n=1}^{S}\sum_{t=1}^{T}\eta\left(\frac{d_t^k}{L_t} \leq \alpha\right)}{S \cdot T} \tag{13}$$

where PCK_k is the k-th joint's PCK. S denotes the number of sequences; T is the number of frames in each sequence. $\eta(\cdot)$ is the indicator function. When the condition in parentheses is true, $\eta(\cdot)$ is 1, otherwise it is 0. ${d_t}^k$ represents euclidean distance between the estimated value and the ground truth. α is the hyperparameter which is set to 0.2 following [11, 12] to control the range of errors. L_t is the threshold of the error distance. According to [11, 12], we set L to the person size and the torso size, respectively. Based on the PCK of each joint, we also use the mean PCK (mPCK) to measure the overall performance of the model [11, 12].

$$mPCK = \frac{\sum_{n=1}^{S}\sum_{t=1}^{T}\sum_{k=1}^{K}\eta\left(\frac{d_t^k}{L_t} \leq \alpha\right)}{S \cdot T \cdot K} \tag{14}$$

4.2. Comparison with State-of-the-arts

4.2.1. Results on the Penn Action dataset

PCK normalized by person size. We record the performance, parameters, and inference time of commonly used approaches, as shown in Table 1. The proposed KIMNet achieves the best performance on the PCK for each joint and mPCK The KIMNet outperforms the LSTMPM [5], TCE [10], DKD [11] and RPSTN [12] by 2.0%, 1.7%, 1.9%, and 1.0% mPCK, respectively. These methods aggregate features along the temporal dimension but ignore the temporal correlations between joints. In contrast to these methods, the proposed KIMNet explicitly models the temporal correlations between joints, which is beneficial for locating joints. Furthermore, compared with the CNN-based methods, such as RPSTN [12] and DKD [11], the attention mechanism makes KMM maintain the high resolution of feature maps, which can avoid the loss of spatial information caused by the reduction of the resolution. Besides, KIMNet abandons the cumbersome convolutional module that is used to model the temporal features of poses, so it spends less time estimating poses during inferencing.

PCK normalized by torso size. As shown in Table 1, our approach also ranks first among the existing methods with fewer parameters and inferencing time. And the overall performance of KIMNet is 1.2% higher than that of

Table 1: Comparisons with the state-of-the-art methods on the Penn Action dataset. Values in brackets indicate the gap between the corresponding approach and the proposed KIMNet.

Methods	Params (M)	Time (ms)	Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	mPCK
			Normalized by person size							
Thin-slicing [8]	-	-	98.0	97.3	95.1	94.7	97.1	97.1	96.9	96.5 (↓ 3.2)
K-FPN [18]	-	-	98.7	98.7	97.0	95.3	98.8	98.7	98.6	98.0 (↓ 1.7)
TCE [10]	-	-	99.3	98.5	97.6	97.2	98.6	98.1	97.4	98.0 (↓ 1.7)
UniPose [6]	-	-	-	-	-	-	-	-	-	99.3 (↓ 0.4)
DCPose [39]	-	-	-	98.6	96.2	96.0	98.7	98.8	98.7	97.9 (↓ 1.8)
LSTMPM [5]	231.5	25	98.9	98.6	96.6	96.6	98.2	98.2	97.5	97.7 (↓ 2.0)
DKD [11]	219.92	11	98.8	98.7	96.8	97.0	98.2	98.1	97.2	97.8 (↓ 1.9)
RPSTN [12]	222.17	12	99.0	98.7	98.8	98.5	98.8	98.7	98.8	98.7 (↓ 1.0)
KIMNet	**214.70**	**10**	**99.4**	**99.6**	**99.1**	**99.0**	**99.8**	**99.5**	**99.4**	**99.7**
			Normalized by torso size							
LSTMPM [5]	231.5	25	96.0	93.6	92.4	91.1	88.3	94.2	93.5	92.6 (↓ 4.3)
DKD [11]	219.92	11	96.6	93.7	92.9	91.2	88.8	94.3	93.7	92.9 (↓ 4.0)
RPSTN [12]	222.17	12	98.2	96.9	95.2	93.2	96.6	95.7	95.0	95.7 (↓ 1.2)
KIMNet	**214.70**	**10**	**98.6**	**97.7**	**96.6**	**95.7**	**96.9**	**97.3**	**96.3**	**96.9**

RPSTN, which demonstrates the effectiveness of the KIMNet. In particular, we obtain encouraging improvements for those more challenging joints, such as *wrists* and *ankles*: with a PCK of 95.7% (↑ 1.8%) for *wrists* and a PCK of 96.3% (↑ 1.5%) for *ankles*, confirming the advantages of the KIMNet in locating the challenging joints. Experimental results prove that the temporal dependency between joints is beneficial for estimating poses from videos.

Compatibility of KMM. We integrate KMM into various pose estimation frameworks, including LSTMPM [5], DKD [11], and RPSTN [12], to evaluate its generality, as shown in Table 2. Specifically, we replace the temporal modeling modules in raw backbones with the KMM. The performance of some joint points decreases slightly after integrating KMM, but the overall performance of the network is improved, which proves that the proposed KMM is compatible with the current popular video-based pose estimation frameworks.

Table 2: Verification of KMM's generality on the Penn Action dataset. Evaluation metric is the PCK normalized by torso size.

Methods	Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	mPCK
LSTMPM [5]	96.0	93.6	**92.4**	91.1	88.3	94.2	**93.5**	92.6 (↓ 0.4)
LSTMPM-KMM	**97.3**	**94.3**	92.3	**93.9**	**89.8**	**94.5**	91.3	**93.0**
DKD [11]	96.6	93.7	**92.9**	91.2	88.8	94.3	**93.7**	92.9 (↓ 0.8)
DKD-KMM	**97.7**	**96.4**	91.5	**96.2**	**89.7**	**95.4**	91.4	**93.7**
RPSTN [12]	**98.2**	96.9	**95.2**	93.2	96.6	95.7	95.0	95.7 (↓ 0.2)
RPSTN-KMM	97.9	**97.0**	94.8	**93.3**	96.6	**96.9**	**95.9**	**95.9**

4.2.2. Results on the Sub-JHMDB dataset

We evaluate the KIMNet on the challenging Sub-JHMDB dataset to further confirm its effectiveness, as shown in Table 3.

PCK normalized by person size. We observe that previous approaches have achieved impressive results on the Sub-JHMDB dataset. KIMNet achieves 98.5% mPCK that surpasses RPSTN [12] by 1.1% and outperforms SLT-Pose [28] by 2.5%. KIMNet explicitly models the temporal dependency between joints across different frames. In this way, KIMNet can locate the current joint via other joints correlated with the current joint. SLT-Pose [28] models the temporal correlation between poses in different frames, but it only considers the pose-level correlations while ignoring the temporal correlation between joints.

PCK normalized by torso size. Similar to the results on the Penn Action dataset, KIMNet also achieves the best performance. The advantages of our model over RPSTN [12] are more significant, achieving over 2.3% mPCK improvement. Especially for the joints with high flexibility, such as *elbows*, *wrists*, *knees*, and *ankles*, our model achieves 5.0%, 5.3%, 2.9%, and 2.4% improvements over the RPSTN [12]. The reason is that KIMNet can explicitly model the temporal dependency between joints across frames while maintaining the original resolution of the feature maps, which makes KIMNet locate one joint according to rich auxiliary information of other joints. However, KIMNet shows inferior performance for the head and hip joints against RPSTN [12]. The possible reason is that a person's scale in the Sub-JHMDB dataset is small. Furthermore,

the head and hip move smoothly relative to other joints. It is possible to locate these two joints through their spatial and temporal features. RPSTN [12] adopts a CNN-based structure to model the dynamics of poses. Compared to the KMM containing two 1x1 convolutions, the CNN-based module can extract more discriminative features about the head and hip.

Table 3: Comparisons with the state-of-the-art methods on the Sub-JHMDB dataset. Values in brackets indicate the gap between the corresponding approach and the proposed KIMNet.

Methods	Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	mPCK
	Normalized by person size							
Thin-slicing [8]	97.1	95.7	87.5	81.6	98.0	92.7	89.8	92.1 (↓ 6.4)
K-FPN [18]	95.1	96.4	95.3	91.3	96.3	95.6	92.6	94.7 (↓ 3.8)
TCE [10]	99.3	98.9	96.5	92.5	98.9	97.0	93.7	96.5 (↓ 2.0)
LSTMPM [5]	98.2	96.5	89.6	86.0	98.7	95.6	90.9	93.6 (↓ 4.9)
DKD [11]	98.3	96.6	90.4	87.1	99.1	96.0	92.9	94.0 (↓ 4.5)
DCPose [39]	-	97.9	93.2	92.1	98.4	97.8	95.9	95.8 (↓ 2.7)
FAMI-Pose [20]	**99.3**	98.6	94.5	91.7	**99.2**	91.8	95.4	96.0 (↓ 2.5)
SLT-Pose [28]	**99.3**	98.6	94.3	91.4	**99.2**	91.9	95.3	96.0 (↓ 2.5)
RPSTN [12]	98.9	**99.1**	**99.0**	97.9	97.8	97.8	**97.3**	97.4 (↓ 1.1)
KIMNet (Ours)	98.9	**99.1**	98.7	**98.5**	98.6	**98.3**	97.2	**98.5**
	Normalized by torso size							
LSTMPM [5]	92.7	75.6	66.8	64.8	78.0	73.1	73.3	73.6 (↓ 14.5)
DKD [11]	**94.4**	78.9	69.8	67.6	81.8	79.0	78.8	77.4 (↓ 10.7)
RPSTN [12]	91.0	87.1	82.1	80.5	**88.8**	85.9	83.8	85.8 (↓ 2.3)
KIMNet (Ours)	90.1	**88.8**	**87.1**	**85.8**	87.9	**88.8**	**86.2**	**88.1**

4.3. Ablation Studies

We first evaluate the effectiveness of the proposed method from the temporal modeling method. Then we change the internal structure and the resolution of KMM to verify the validity of KMM. Finally, we validate the influence of the feature encoder.

4.3.1. Ablation studies about temporal modeling method

To demonstrate the effectiveness of the KMM, we compare the proposed KMM with previous temporal modeling modules, as shown in Table 4. We first

remove the temporal modeling module, *i.e.*, the Baseline. Obviously, compared with our KIMNet, the model's performance degrades significantly after removing the temporal modeling, which proves the necessity of temporal modeling for video-based human pose estimation.

Second, to evaluate the predictive performance of KMM, we remove the features for frame $t+1$ (*i.e.*, $F(I_{t+1})$ in eq. 2, and denote it as "Predicted" in Table 4). Experimental results show that KMM provides acceptable prediction results. After integrating the observations of the current frame, the model obtains accurate positions of joints, demonstrating the necessity of $F(I_{t+1})$ for refining joints' positions.

Third, we adopt the DKD proposed in [11] and JRPSP introduced by [12] to model the temporal information. Specifically, we successively replace the KMM in KIMNet with DKD and JRPSP. Compared with DKD and JRPSP, KMM brings a 1.1% and 0.9% improvement of overall performance. Especially for the flexible *wrists* and *ankles*, the improvements are obvious. These joints generally move violently and are easily prone to motion blur. It is helpful to locate these joints by fusing other joints' information.

Table 4: Ablation studies about the temporal modeling module. Evaluation metric is the PCK normalized by torso size. Values in brackets indicate the gap between the corresponding approach and our method.

Method	Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	mPCK
Baseline	97.5	96.7	93.1	91.2	96.0	95.1	93.7	94.5 (↓ 2.4)
Predicted	93.0	94.5	92.3	85.1	95.2	90.0	89.3	91.2 (↓ 5.7)
DKD [11]	97.2	97.0	94.6	93.6	96.3	96.8	95.4	95.8 (↓ 1.1)
JRPSP [12]	97.5	97.2	96.2	92.5	97.1	96.9	95.5	96.0 (↓ 0.9)
KIMNet	**98.6**	**97.7**	**96.6**	**95.7**	**96.9**	**97.3**	**96.3**	**96.9**

4.3.2. Ablation studies about KMM's structure

We have carried out a comprehensive analysis about the structure of KMM. Experimental results are recorded in Table 5.

Influence of shared $Conv_h$ and $Conv_c$. We adopt two 1×1 convolutions

whose weights are shared to extract pose representations from two adjacent frames, *i.e.*, KIMNet-Shared-hc in Table 5. As can be seen that the performance of KIMNet-Shared-hc decreases by 1.1% mPCK over that of KIMNet. Two independent convolutions can extract discriminative pose features from two contiguous frames, which is beneficial for capturing the robust motion information of joints.

Influence of historical heatmaps. We introduce the residual, *i.e.*, heatmaps in frame t, into KMM for evaluating the influence of historical position for predicting joints' positions, *i.e.*, KIMNet-Residual. The performance decreases obviously after introducing the historical position of joints. We think that m^p_{t+1} has included the positional information about joints in frame $(t+1)$. The model pays much attention to the historical position after introducing historical positions.

Temporal correlations between heatmaps. We adopt heatmaps in two frames to model the temporal correlation between joints, *i.e.*, KIMNet-Heatmap. The performance of KIMNet-Heatmap also decreases a lot. Compared with joints' heatmaps which include positional information, pose features f contain rich semantic information about joints, including appearance features and positional information, which is helpful for learning the robust motion cues.

Table 5: Ablation studies about the rationality of KMM's design. Evaluation metric is the PCK normalized by torso size. Values in brackets indicate the gap between the corresponding approach and our method.

Method	Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	mPCK
KIMNet-Share-hc	97.4	97.1	95.0	93.1	96.8	96.7	95.1	95.8 (↓ 1.1)
KIMNet-Residual	97.4	95.7	88.9	81.1	94.8	91.8	85.9	90.3 (↓ 6.6)
KIMNet-Heatmap	96.2	94.9	86.3	82.5	95.6	92.8	87.6	90.4 (↓ 6.5)
KIMNet	**98.6**	**97.7**	**96.6**	**95.7**	**96.9**	**97.3**	**96.3**	**96.9**

4.3.3. Ablation studies about resolution of propagating features

The attention mechanism allows KMM to maintain the original resolution of pose features during modeling the temporal dependency between joints. We

gradually reduce the resolution of KMM's input to verify the importance of retaining the original resolution during propagating pose knowledge. As shown in Table 6, the performance of the KIMNet degrades with the reduction of the resolution, demonstrating that the significance of the high resolution for pose features.

Table 6: Ablation studies about the resolution of KMM's input. Evaluation metric is the PCK normalized by torso size. "KIMNet-S" represents the size of KMM's input is $S \times S$. Values in brackets indicate the gap between the corresponding approach and our method.

Method	Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	mPCK
KIMNet-8	96.0	94.8	88.2	81.1	95.9	93.1	88.9	90.8 (↓ 6.1)
KIMNet-16	96.6	95.3	90.9	85.8	96.6	94.6	90.5	92.4 (↓ 4.5)
KIMNet-32	97.3	95.8	91.0	87.0	97.2	96.1	91.7	93.5 (↓ 3.4)
KIMNet-64	**98.6**	**97.7**	**96.6**	**95.7**	**96.9**	**97.3**	**96.3**	**96.9**

4.3.4. Ablation studies about feature encoder

We adopt various models, including SimpleBaseline [15] with different structures (ResNet-18, ResNet-34, and ResNet-50), Hourglass [14], and HRNet [16] as the feature encoder to evaluate the influence of different backbones. Feature encoders are pre-trained on the MPII dataset [37]. Experimental results are listed in Table 7. For different feature encoders, the performance of KIMNet is improved with the enhancement of the feature extraction ability of the feature encoder. Compared to the SimpleBaseline with ResNet-50 as the backbone, *i.e.*, KIMNet†, Hourglass [14] (KIMNet(HG)) and HRNet (KIMNet(HRNet)) bring limited improvement. Considering the performance and computational cost, the SimpleBaseline with ResNet-50 as the backbone is chosen as the feature encoder in this paper.

4.4. Qualitative Analysis about Features Learned by KMM

4.4.1. Visualization of temporal correlation

To intuitively observe the temporal dependency between joints at different times, we randomly visualize two continuous frames and the corresponding

Table 7: Ablation studies about the feature encoder. Evaluation metric is the PCK normalized by torso size. KIMNet† represents the structure used in this paper. Values in brackets indicate the gap between the corresponding approach and the proposed KIMNet†.

Methods	Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	mPCK
KIMNet (Res18)	97.2	97.1	95.0	93.9	96.1	96.2	94.8	95.6 (↓ 1.3)
KIMNet (Res34)	97.6	97.3	95.4	93.3	96.2	96.4	95.0	95.8 (↓ 1.1)
KIMNet (HG)	98.4	97.7	96.8	95.6	97.2	**97.6**	**97.1**	97.1 (↑ 0.2)
KIMNet (HRNet)	98.0	**97.8**	**97.2**	**95.7**	**97.3**	97.5	**97.1**	**97.2** (↑ 0.3)
KIMNet† (Res50)	**98.6**	97.7	96.6	**95.7**	96.9	97.3	96.3	96.9

temporal correlation matrix, as shown in Figure4. The size of joints at frame t represents the degree of temporal dependency. The *right wrist* has a high temporal dependency with the *right wrist*, *right elbow*, and *shoulders* in frame t. At the same time, the *left* and *right ankles* work as the support foot and the force generating foot, respectively, and they cooperate with the *right wrist* to complete the action while ensuring the coordination of the action. Therefore, there is a temporal dependency between the *right wrist* and two *ankles* in frame t. These joints provide extra temporal information to help locate the *right wrist* from frame $t+1$. Similarly for the occluded *left elbow* at frame $t+1$, the auxiliary temporal information provided by the *head*, *shoulders*, *wrists*, and *left knee* at frame t helps the model to locate *left elbow* from frame $t+1$. Therefore, with the help of the temporal dependency between joints, the proposed KIMNet uses more extra temporal information provided by other joints to locate the current joint rather than only depending on the current joint's information to locate it.

4.4.2. Visualization of joint heatmaps

To further validate the effectiveness of KIMNet, we visualize the predicted maps of KMM and heatmaps of KIMNet for different poses with various complexity, and display heatmaps of ground truth (2^{nd} column), outputs of KMM (the 3^{rd} column), and KIMNet (the 4^{th} column) in Figure 5. Specifically, we randomly select one frame from the 2^{nd} frame to the T-th frame and visualize heatmaps of joints.

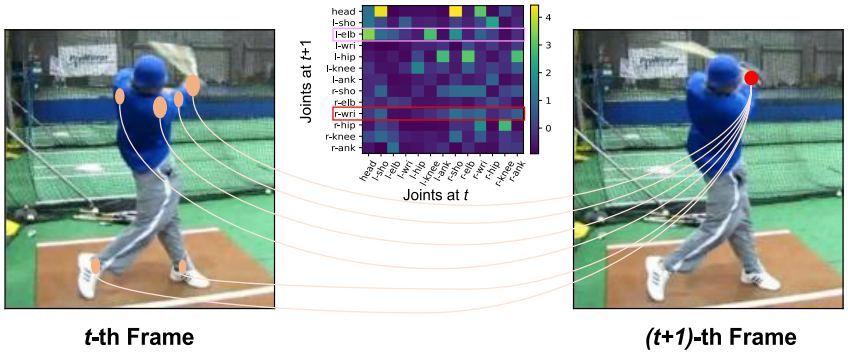


Figure 4: Visualization of the temporal correlation between joints. The size of joints at frame t represents the degree of temporal dependency between joints.

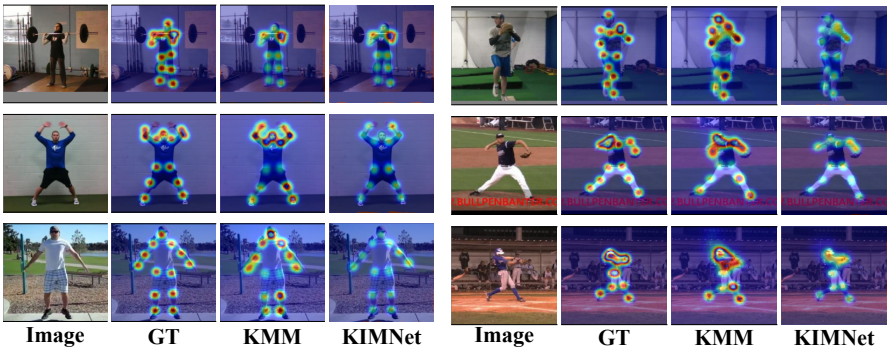
We observe that KMM can provide accurate initial positions of joints for those simple actions, as shown in Figure 5a, actions *weight lifting* and *jumping jack*. Joints move homogeneously in these simple actions. Thus, it is easy for KMM to model the temporal dependency between joints, which makes KMM learn obvious motion cues for these joints. Moreover, KMM can guide the model to pay much attention to the posing area for those complex actions, as shown in Figure 5b. For the action *play baseball*, joints move violently, especially the joints of the upper limbs. Therefore, KMM pays much attention to the upper limb, indicating that joints of the upper limbs have similar movements. Based on the regions activated by the KMM, KIMNet can locate joints accurately.

4.5. Verification of Performance in Occlusion Scenes

To explore the potential of the proposed KIMNet in occluded scenes, we evaluate the KIMNet in two cases of the occlusion, including the manually occluded scenario and the naturally occluded scenario.

4.5.1. The Manually Occluded Scenario

We randomly generate several masks with a size of 40×40 from time and space to occlude joints for evaluating KIMNet's performance against the temporal and spatial occlusion, respectively.



(a) Visualization for simple poses. (b) Visualization for complex poses.

Figure 5: Outputs of the proposed KMM and KIMNet. The visualization results from left to right are the original input frames, ground truth, outputs of the KMM, and outputs of the KIMNet.

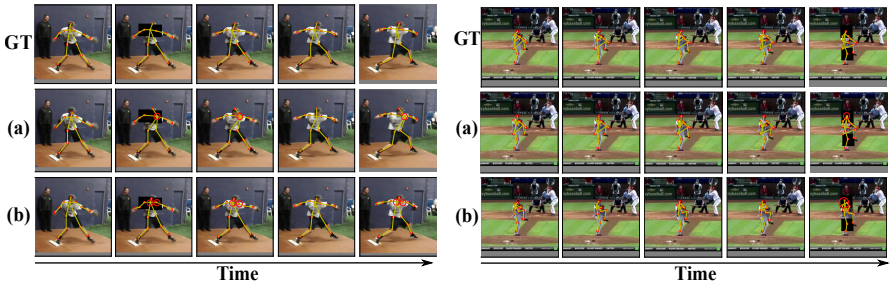


Figure 6: Visualization of the temporal occlusion. (a) is the output of RPSTN [12]. (b) is the output of KIMNet. We use red circles to highlight the joints that are localized more accurate by KIMNet.

Verification of temporal occlusion. We randomly select a frame from the input sequence and generate two masks to occlude joints. Experimental results of the proposed KIMNet and RPSTN are shown in Figure6. Because our proposed KIMNet explicitly models the temporal dependency between joints across different frames, it can use all joints in the previous frame to help locate joints in the current frame.

Verification of spatial occlusion. To explore the potential of KIMNet for occlusion, we randomly generate two masks from the 2^{nd} to the T-th frame, as shown in Figure7. Because the spatial information of the occluded pose is missing, it is difficult for RPSTN to match similar regions in adjacent frames. Therefore, RPSTN provides inferior results for occluded joints. The proposed KIMNet can utilize all joints in the 1^{st} frame to locate occluded joints. It can provide reasonable positions for occluded joints in subsequent frames.

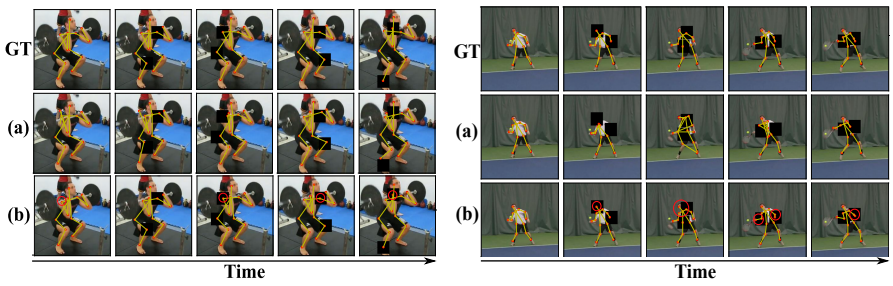


Figure 7: Visualization of occluded poses. (a) is the output of RPSTN [12]. (b) is the output of KIMNet. We use red circles to highlight the joints that are localized more accurate by KIMNet.

Influence of the number of occluded joints. To evaluate the influence of the number of occluded joints, the number of occluded joints is increased from 1 to 9. Under the same experimental settings, we compare the performance of the proposed KIMNet with existing methods, as shown in Figure8. With the increase in the number of occluded joints, the performance of all models declines. However, compared to previous approaches, our KIMNet is more robust to occlusion.

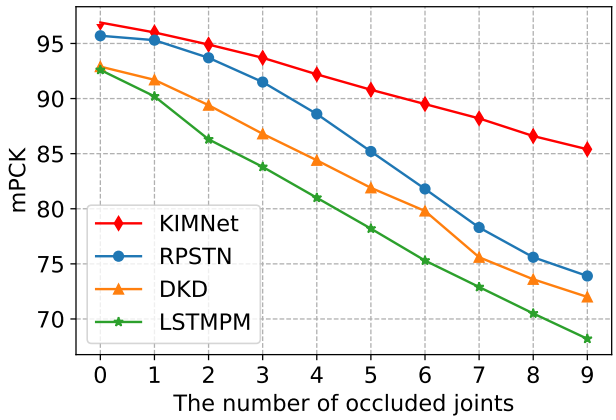


Figure 8: Comparison of the performance between KIMNet and existing methods under the occlusion scene.

4.5.2. The Naturally Occluded Scenario

To explore the potential of the KIMNet, we evaluate it on the naturally occluded scenario. Qualitative results are shown in Figure 9. Because the KIMNet considers the temporal dependency between the joint in the current frame and all joints in the previous frame, the KIMNet is able to locate the current joint according to the information of all joints in the previous frame. Therefore, the proposed KIMNet provides more reasonable positions of the occluded joints (such as the joints in red circles).

5. Conclusion

Capturing temporal correlations between joints at different time is the crucial issue for video-based pose estimation. In this paper We propose a kinematics modeling network (KIMNet) that equipped with a plug-and-play kinematics modeling module (KMM) to capture each joint's motion cues by modeling the temporal correlation between any two joints across frames. In this way, the model can locate the current joint according to all joints' information in the previous frame, rather than only relying on the current joint's information. Especially, for the joints suffering from the occlusion or motion blur, information from other

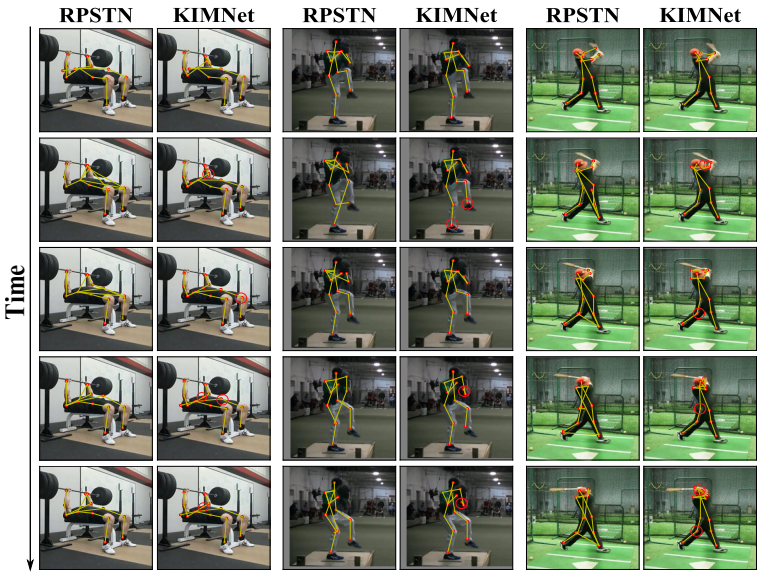


Figure 9: Experimental results in naturally occluded scenario. Joints that the KIMNet estimates better than the RPSTN are marked by the red circles.

joints is beneficial for locating them. Experimental results show the advantages of the proposed KIMNet in achieving state-of-the-art results on two challenging benchmarks.

Limitation and Discussion. Although the proposed KIMNet performs well against state-of-the-arts on the challenging datasets, it also suffers from some dilemmas in extremely complex scenarios, such as the severely occluded or truncated poses. For the heavily occluded poses, almost half of the joints are invisible, making it difficult to locate those invisible joints through several visible joints. In this case, one possible solution is that the symmetry of the human body can be used to help locate the joints on the occluded side of the body. For the severely truncated poses, the structural information of human poses is destroyed to a certain extent, which makes it challenging for the model to learn the correlation between joints. In this case, the possible solution is to introduce the prior knowledge of the human body to assist model learning. Therefore, we will do further research from above two aspects in future work.

Acknowledgments

This work was supported partly by the National Natural Science Foundation of China (Grant No. 62173045, 61673192), partly by the Fundamental Research Funds for the Central Universities (Grant No. 2020XD-A04-3), and the Natural Science Foundation of Hainan Province (Grant No. 622RC675).

References

[1] L. G. Foo, T. Li, H. Rahmani, Q. Ke, J. Liu, Unified pose sequence modeling, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2023, pp. 13019–13030.

[2] V. Mazzia, S. Angarano, F. Salvetti, F. Angelini, M. Chiaberge, Action transformer: A self-attention model for short-time pose-based human action recognition, Pattern Recognit. 124 (2022) 108487.

[3] H. Wang, L. Wang, Learning content and style: Joint action recognition and person identification from human skeletons, Pattern Recognit. 81 (2018) 23–35.

[4] J. He, J. Sun, Q. Liu, S. Peng, Nrpose: Towards noise resistance for multi-person pose estimation, Pattern Recognit. 142 (2023) 109680.

[5] Y. Luo, J. S. J. Ren, Z. Wang, W. Sun, J. Pan, J. Liu, J. Pang, L. Lin, LSTM pose machines, in: Proc. IEEE Conference Computer Vision and Pattern Recognition (CVPR), 2018, pp. 5207–5215.

[6] B. Artacho, A. E. Savakis, Unipose: Unified human pose estimation in single images and videos, in: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 7033–7042.

[7] T. Pfister, J. Charles, A. Zisserman, Flowing convnets for human pose estimation in videos, in: Proc. IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1913–1921.

[8] J. Song, L. Wang, L. V. Gool, O. Hilliges, Thin-slicing network: A deep structured model for pose estimation in videos, in: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 5563–5572.

[9] M. Wang, J. Tighe, D. Modolo, Combining detection and tracking for human pose estimation in videos, in: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 11085–11093.

[10] Y. Li, K. Li, X. Wang, R. Y. D. Xu, Exploring temporal consistency for human pose estimation in videos, Pattern Recognit. 103 (2020) 107258.

[11] X. Nie, Y. Li, L. Luo, N. Zhang, J. Feng, Dynamic kernel distillation for efficient pose estimation in videos, in: Proc. IEEE International Conference on Computer Vision (ICCV), 2019, pp. 6941–6949.

[12] Y. Dang, J. Yin, S. Zhang, Relation-based associative joint location for human pose estimation in videos, IEEE Trans. Image Process. 31 (2022) 3973–3986.

[13] P. Ding, J. Yin, Towards more realistic human motion prediction with attention to motion coordination, IEEE Trans. Circuits Syst. Video Technol. 32 (9) (2022) 5846–5858.

[14] A. Newell, K. Yang, J. Deng, Stacked hourglass networks for human pose estimation, in: B. Leibe, J. Matas, N. Sebe, M. Welling (Eds.), Proc. European Conference Computer Vision (ECCV), 2016, pp. 483–499.

[15] B. Xiao, H. Wu, Y. Wei, Simple baselines for human pose estimation and tracking, in: Proc. European Conference Computer Vision (ECCV), 2018, pp. 472–487.

[16] K. Sun, B. Xiao, D. Liu, J. Wang, Deep high-resolution representation learning for human pose estimation, in: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 5693–5703.

[17] Y. Bin, Z. Chen, X. Wei, X. Chen, C. Gao, N. Sang, Structure-aware human pose estimation with graph convolutional networks, Pattern Recognit. 106 (2020) 107410.

[18] Y. Zhang, Y. Wang, O. I. Camps, M. Sznaier, Key frame proposal network for efficient pose estimation in videos, in: Proc. European Conference Computer Vision (ECCV), 2020, pp. 609–625.

[19] R. Feng, Y. Gao, T. H. E. Tse, X. Ma, H. J. Chang, Diffpose: Spatiotemporal diffusion model for video-based human pose estimation, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2023, pp. 14861–14872.

[20] Z. Liu, R. Feng, H. Chen, S. Wu, Y. Gao, Y. Gao, X. Wang, Temporal feature alignment and mutual information maximization for video-based human pose estimation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 11006–11016.

[21] J. Yu, Y. Rui, D. Tao, Click prediction for web image reranking using multimodal sparse coding, IEEE Trans. Image Process. 23 (5) (2014) 2019–2032.

[22] J. Zhang, Y. Cao, Q. Wu, Vector of locally and adaptively aggregated descriptors for image feature representation, Pattern Recognit. 116 (2021) 107952.

[23] L. Tian, P. Wang, G. Liang, C. Shen, An adversarial human pose estimation network injected with graph structure, Pattern Recognit. 115 (2021) 107863.

[24] C. Jiang, K. Huang, S. Zhang, X. Wang, J. Xiao, Y. Goulermas, Aggregated pyramid gating network for human pose estimation without pre-training, Pattern Recognit. 138 (2023) 109429.

[25] S. Yang, Z. Feng, Z. Wang, Y. Li, S. Zhang, Z. Quan, S. Xia, W. Yang, Detecting and grouping keypoints for multi-person pose estimation using instance-aware attention, Pattern Recognit. 136 (2023) 109232.

[26] T. Wang, X. Zhang, Gated region-refine pose transformer for human pose estimation, Neurocomputing 530 (2023) 37–47.

[27] Y. Wang, Y. Luo, G. Bai, J. Guo, Uformpose: A u-shaped hierarchical multi-scale keypoint-aware framework for human pose estimation, IEEE Trans. Circuits Syst. Video Technol. 33 (4) (2023) 1697–1709.

[28] D. Gai, R. Feng, W. Min, X. Yang, P. Su, Q. Wang, Q. Han, Spatiotemporal learning transformer for video-based human pose estimation, IEEE Trans. Circuits Syst. Video Technol. 33 (9) (2023) 4564–4576.

[29] L. Wu, Y. Wang, L. Shao, M. Wang, 3-d personvlad: Learning deep global representations for video-based person reidentification, IEEE Trans. Neural Networks Learn. Syst. 30 (11) (2019) 3347–3359.

[30] W. Wang, T. Zhou, F. Yu, J. Dai, E. Konukoglu, L. V. Gool, Exploring cross-image pixel contrast for semantic segmentation, in: Procedings of the IEEE International Conference on Computer Vision (ICCV), 2021, pp. 7283–7293.

[31] G. Sun, W. Wang, J. Dai, L. V. Gool, Mining cross-image semantics for weakly supervised semantic segmentation, in: Procedings of the European Conference Computer Vision (ECCV), 2020, pp. 347–365.

[32] X. Lu, W. Wang, C. Ma, J. Shen, L. Shao, F. Porikli, See more, know more: Unsupervised video object segmentation with co-attention siamese networks, in: Procedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 3623–3632.

[33] X. Lu, W. Wang, J. Shen, D. J. Crandall, J. Luo, Zero-shot video object segmentation with co-attention siamese networks, IEEE Trans. Pattern Anal. Mach. Intell. 44 (4) (2022) 2228–2242.

[34] J. Wu, J. Cao, L. Song, Y. Wang, M. Yang, J. Yuan, Track to detect and segment: An online multi-object tracker, in: Procedings of the IEEE

Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 12352–12361.

[35] M. W. Lee, R. Nevatia, Dynamic human pose estimation using markov chain monte carlo approach, in: Procedings of the IEEE Workshop on Applications of Computer Vision (WACV), 2005, pp. 168–175.

[36] H. Xu, J. Zhang, J. Cai, H. Rezatofighi, D. Tao, Gmflow: Learning optical flow via global matching, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 8121–8130.

[37] M. Andriluka, L. Pishchulin, P. V. Gehler, B. Schiele, 2d human pose estimation: New benchmark and state of the art analysis, in: Procedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 3686–3693.

[38] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: Procedings of the International Conference on Learning Representations (ICLR), 2015.

[39] Z. Liu, H. Chen, R. Feng, S. Wu, S. Ji, B. Yang, X. Wang, Deep dual consecutive network for human pose estimation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 525–534.