Human Motion Prediction Using Adaptable Recurrent Neural Networks and Inverse Kinematics

Ruixuan Liu, *Graduate Student Member, IEEE*, and Changliu Liu, *Member, IEEE*

***Abstract*—Human motion prediction, especially arm prediction, is critical to facilitate safe and efficient human-robot collaboration (HRC). This letter proposes a novel human motion prediction framework that combines a recurrent neural network (RNN) and inverse kinematics (IK) to predict human arm motion. A modified Kalman filter (MKF) is applied to adapt the model online. The proposed framework is tested on collected human motion data with up to 2 s prediction horizon. The experiments demonstrate that the proposed method improves the prediction accuracy by approximately 14% comparing to the state-of-art on seen situations. It stably adapts to unseen situations by keeping the maximum prediction error under 4 cm, which is 70% lower than other methods. Moreover, it is robust when the arm is partially occluded. The wrist prediction remains the same, while the elbow prediction has 20% less variation.**

***Index Terms*—Adaptive systems, human-in-the-loop control, kalman filtering.**

I. Introduction

The rapid development of human-robot collaboration (HRC) addresses contemporary needs by enabling more efficient and flexible production lines. Working in such environments, human workers collaborate with robot arms in a confined workspace. Therefore, it is essential to ensure safety while maximizing efficiency.

Human motion prediction is an important step to address the challenge [1]. However, motion prediction is difficult due to the stochastic and nonlinear nature of human behaviors. Early approaches [2] addressed the problem in a probabilistic way by using Hidden Markov Models to estimate the possible areas that human arms are likely to occupy. Assuming human motions are optimal, [3] intended to learn a cost function of human behaviors by inverse optimal control and made prediction according to the learned cost function. Recent works [4], [5] addressed the prediction as a reaching problem by specifically learning the motion of human hands using neural networks. The recent development of recurrent neural network (RNN) had outstanding performance in motion prediction [6]. The Encoder-Recurrent-Decoder (ERD) structure [7], which transformed the joint angles to higher-dimensional features, was shown to be effective in motion prediction. Reference [8] added the sequence-to-sequence architecture to address the prediction as a machine translation problem. Recently, [9], [10] devoted to embedding the structural information of human bodies into the neural networks. However, existing methods suffer from several problems. First, neural networks are pre-trained and fixed, which may have limited generalizability or adaptability to unseen situations. Second, the physical constraints of the human body are encoded using complex neural network structures, which is unintuitive and difficult to verify.

This letter focuses on human arm motion since the major body parts involved in the collaboration on production lines are arms. In general, hand motion carries contextual information such as intentions, while the elbow motion mainly supports the hand motion. Therefore, we divide the problem into two subtasks: wrist prediction and full-arm prediction. This letter proposes a novel RNNIK-MKF arm motion prediction framework, which uses an RNN to predict the wrist motion and inverse kinematics (IK) to predict the full-arm motion according to the wrist prediction. A modified Kalman filter (MKF) is used for adapting the model online. This decomposed approach is related to planning-based prediction methods [11], [12]. Nonetheless, this letter is the first to decompose arm prediction models.

The proposed method has several advantages. First, our method uses 3D positions of human arms, hence can be easily deployed in the real world. Many current approaches, such as [7], [13], consider the joint angles or Electromyography signals generated by muscles as inputs. However, these measurements either require extra transformation steps or specialized sensors, making them inconvenient and inefficient to be applied to real applications. The 3D positions are easier to be captured by regular sensors, such as depth cameras. Second, the method is adaptable, hence can easily generalize to unseen situations. Many existing approaches have been proven to work well in trained environments, but have limited generalizability. However, it is impossible to obtain motion data from all workers and comprehensively validate the model for all situations. The proposed method enables online model adaptation using MKF to achieve better generalizability. Third, the proposed structure can explicitly encode the kinematic constraints into the prediction model. Current

Manuscript received August 31, 2020; revised November 5, 2020; accepted November 25, 2020. Date of publication December 4, 2020; date of current version December 28, 2020. This work was supported in part by Ford Motor Company. Recommended by Senior Editor M. Guay. *(Corresponding author: Changliu Liu.)*

The authors are with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213 USA (e-mail: ruixuanl@andrew.cmu.edu; cliu6@andrew.cmu.edu).

Digital Object Identifier 10.1109/LCSYS.2020.3042609

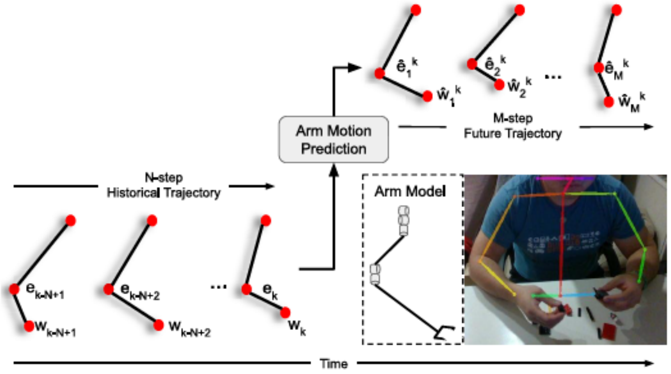


Fig. 1. Illustration of the arm motion prediction problem. The problem takes in the N-step historical arm trajectory and outputs the M-step future arm trajectory. The human arm is modeled as a 5-DOF manipulator.

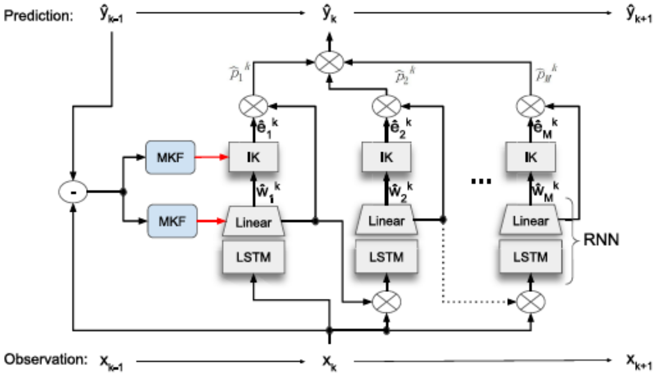


Fig. 2. RNNIK-MKF Prediction Framework.

approach, such as [9], used complex neural network structures to encode the structural information, which is complicated and implicit. The proposed method uses the physical model derived from the human arm, which is simple and intuitive. Fourth, the proposed method is robust when occlusion of the arm happens. Few existing methods [14] considered the situation when a partial human body is not observable. Yet the situation is common during collaboration as the robot could go between the sensor and the human arm, and thus, occluding the body part. Our method using the physical arm model can robustly predict the arm positions even when a portion of the arm, e.g., elbow, is blocked from view. The proposed method is tested on collected human motion data. The results demonstrate that our method outperforms the existing methods with the advantages mentioned above. Our code is available at github.com/intelligent-control-lab/RNNIK-MKF.

The remainder of the letter is organized as follows. Section II formulates the motion prediction problem. Section III proposes the RNNIK-MKF method. Section IV compares RNNIK-MKF with several existing methods. Section V concludes the letter.

II. Problem Formulation

This letter tackles the prediction problem on production lines, where humans collaborate with robot arms in confined workspaces. In these environments, the trunk of the human body tends to stay still and the major body parts interacting with the robot are the human arms. Therefore, we focus on predicting human arm motion, elbow and wrist, as shown in Fig. 1. This letter discusses single-arm prediction, but the methodology can be easily extended to dual-arm prediction.

This letter uses regular symbols and $\wedge$ to denote the observation and prediction respectively. The wrist and elbow positions at time k are denoted as $w_k,\ e_k \in \mathbb{R}^3$. The position vector $p \in \mathbb{R}^6$ is constructed by stacking the positions of wrist and elbow. We define the observation and prediction at time k as $x_k = [p_{k-N+1}; \dots; p_{k-1}; p_k] \in \mathbb{R}^{6N}$ and $\hat{y}_k = [\hat{p}_1^k; \hat{p}_2^k; \dots; \hat{p}_M^k] \in \mathbb{R}^{6M}$ where $[;]$ means vertical concatenation. The constants $M,\ N \in \mathbb{N}$ are the prediction and observation horizons. $\hat{p}_j^k$ is the jth step prediction at time k. We need to build a prediction model, $\hat{y}_k = f(x_k, \hat{y}_{k-1})$, which takes in the observed N-step historical trajectory and the previous M-step prediction and outputs the M-step future trajectory. Due to the stochastic and time-varying nature of human motion, f can be a time-varying function.

III. RNNIK-MKF Motion Prediction

Humans move their hands intentionally while the elbows are mainly to support the hand motion. Hence, we use an RNN for wrist prediction and IK to extend the wrist prediction to full-arm prediction. MKF is used for online adapting the models to different humans and tasks, and address the time-varying nature of human motion. The proposed RNNIK-MKF framework is shown in Fig. 2.

A. RNN for Wrist Motion Prediction

Human motion is complex and highly nonlinear. In addition, the human motion has strong temporal connections. Therefore, we choose RNN since it has hidden states to memorize past information. In particular, the Long Short-term Memory (LSTM) cell [15] is used since it has adequate gates to control the memory either to remember or to forget. We use an N-to-1 structure [7], where the RNN takes in the N-step wrist history and outputs the next step wrist prediction as shown in Fig. 2. For an M-step prediction, the network iteratively appends the new prediction to the input and then predicts for the next step, until the M-step prediction is obtained as shown in (2). The structure has the advantage to enable more flexibility on the online adaptation since we can adapt the model as soon as a new observation is available.

We use *LSTM* and *RNN* to denote the transition of a single LSTM cell and the N-to-1 prediction respectively. Given the N-step historical wrist trajectory at time k, the hidden states of the LSTM cells are propagated as $[h_1,\ c_1] = \mathit{LSTM}(w_{k-N+1},\ 0,\ 0)$, $[h_2,\ c_2] = \mathit{LSTM}(w_{k-N+2},\ h_1,\ c_1)$, $\dots, [h_N,\ c_N] = \mathit{LSTM}(w_k,\ h_{N-1},\ c_{N-1})$. The first step wrist prediction is obtained using a linear layer

$$\hat{w}_1^k = \phi_k h_N + b_l, \tag{1}$$

where ϕ_k is the adaptable weight matrix of the linear layer at time k and b_l is a constant bias. The LSTM cells and the linear layer (1) construct the N-to-1 *RNN*. We solve the prediction problem as

$$\begin{aligned} \hat{w}_1^k &= \mathit{RNN}(\phi_k,\ [w_{k-N+1},\ ,\dots,\ w_k]^T), \\ \hat{w}_2^k &= \mathit{RNN}(\phi_k,\ [w_{k-N+2},\ ,\dots,\ w_k,\ \hat{w}_1^k]^T), \\ &\dots \end{aligned} \tag{2}$$

until M-step wrist prediction is obtained.

B. IK for Arm Motion Prediction

A general human arm can be decomposed into a 5-DOF manipulator as shown in Fig. 1. The shoulder is decomposed into 3 revolute joints and the elbow is decomposed into 2 revolute joints. The wrist is considered as the end-effector. The state of the arm in the joint space is $\theta \in \mathbb{R}^5$. The end-effector state is $w \in \mathbb{R}^3$. The IK problem solves for $\hat{\theta}$ under the predicted wrist position $\hat{w}$ where $\hat{\theta} = IK(\hat{w}|\theta, w)$, given the current states θ and w. A popular approach to solve the IK problem is using the Jacobian [16]. The Jacobian $J(\theta) \in \mathbb{R}^{3\times n}$ for an n-DOF robot manipulator is

$$J(\theta) = \frac{\partial FK(\theta)}{\partial \theta} \approx \left[\frac{\partial IK(w)}{\partial w}\right]^{-1}, \tag{3}$$

where $FK(\theta) : \mathbb{R}^n \to \mathbb{R}^3$ is the forward kinematics. Using matrix transpose to replace matrix inverse, we can solve the IK problem as

$$\hat{\theta} = \theta + AJ(\theta)^{\mathsf{T}}(\hat{w} - w), \tag{4}$$

where $A \in \mathbb{R}^{n\times n}$ is an adaptable parameter matrix that encodes the individual differences on the joint velocities, e.g., some workers tend to place elbow on table, while others prefer to move with wrist. We can solve the arm prediction by solving the IK for each predicted wrist position.

C. Online Adaptation With MKF

Human motion is time-varying. For instance, a worker might initially move the entire arm. But after a while, the worker would probably rest the elbow on the table and only move the wrist. Moreover, different individuals can perform the same task very differently. Therefore, online adaptation is important to make the method robust and generic. In the proposed framework, the wrist predictor has a linear output layer (1) in ϕ, and the arm predictor (4) is linear in A. Therefore, we apply linear adaptation to the system.

Many existing online adaptation algorithms are based on stochastic gradients [17]. However, these methods have no guarantee of optimality. Thus, we use a second-order method in order to achieve better convergence and optimality. Recursive least-squares parameter adaptation algorithm (RLS-PAA) [18] is an optimal method. However, since the regular RLS-PAA does not consider noise models, it is inefficient to tune. In addition, we need to apply smoothing techniques to the internal adaptation parameters to ensure stable adaptation and prediction under noisy measurements. Moreover, we want the more recent information to have more impact on the current estimation. Hence, we propose to use MKF for online adaptation. We add a forgetting factor, λ, to the conventional Kalman filter to prevent the estimation from saturation. Also, we apply the Exponential Moving Average (EMA) filtering method discussed in [19] to smooth the adaptation process.

The linear prediction system can be written as

$$\begin{aligned} \Phi_k &= \Phi_{k-1} + \omega_k, \\ \hat{y}_k &= \Phi_k X_k + v_k. \end{aligned} \tag{5}$$

For wrist prediction, $\hat{y}$ is the wrist prediction. Φ and X are the parameter matrix ϕ and hidden feature h_N in (1). For arm prediction, $\hat{y}$ is the arm prediction, Φ and X are equivalent to A and $J(\theta)^{\mathsf{T}}(\hat{w} - w)$ in (4). $\omega_k \sim N(0, \sigma_w)$ and $v_k \sim N(0, \sigma_v)$ are virtual Gaussian white noises. The MKF adaptation algorithm is summarized in Algorithm 1. There are two internal variables: the covariance matrix Z and the parameter update step V. The learning gain, K, is calculated on line 6 using Kalman filter's formula. V is calculated using the EMA filtering on line 7. The parameter matrix Φ is updated on line 8. Then Z is updated using the forgetting factor on line 9 and then smoothed using EMA on line 10.

Algorithm 1 Modified Kalman Filter

```
1: Input: MKF parameters: $\lambda > 0$, $\sigma_w \ge 0$, $\sigma_v \ge 0$.
2: Input: EMA parameters: $0 \le \mu_v < 1$, $0 \le \mu_p < 1$.
3: Input: $\Phi_{k-1}$, $E_k = (y_{k-1} - \hat{y}_{k-1})$, $X_{k-1}$.
4: Output: $\Phi_k$.
5: Internal State: $Z$, $V$.
6: $K = Z_{k-1}X_{k-1}^T(X_{k-1}Z_{k-1}X_{k-1}^T + \sigma_w I)^{-1}$
7: $V_k = \mu_v V_{k-1} + (1 - \mu_v)KE_k$
8: $\Phi_k = \Phi_{k-1} + V_k$
9: $Z^* = \frac{1}{\lambda}(Z_{k-1} - KX_{k-1}Z_{k-1} + \sigma_u I)$
10: $Z_k = \mu_p Z_{k-1} + (1 - \mu_p)Z^*$
```

Following the approach in [20] that applies RLS-PAA to the linear output layer of a fully connected neural network (FCNN), we apply the MKF to (1) with respect to ϕ as well as to (4) with respect to A. The virtual noises ω and v are tunable in the adaptation.

The proposed RNNIK-MKF framework is then summarized in Algorithm 2. *MKF* denotes the adaptation in Algorithm 1. The superscripts w and a distinguish the variables for wrist and arm predictions. $FK_{ew} = [FK; FK_e]$ where FK and FK_e are the forward kinematics to wrist and elbow. IK_{ew} solves for the arm state θ_{pre} when both wrist and elbow positions are known. For each time step, the algorithm obtains the current configuration in line 6. The parameters are updated using MKF on lines 11, 13. If occlusion happens, the adaptation is turned off on line 16. Then the algorithm iteratively predicts the future wrist and elbow trajectories for M steps on lines 20 and 22.

IV. Experiments and Results

We had humans sitting at a table doing assembly tasks using LEGO pieces as shown in Fig. 1. The experiments had 3 humans doing 5 assembly tasks for 3 trials each. Each task required 4 LEGO pieces. For each trial, the human was given a set of usable pieces on the table and pictures of the desired assembled object. The duration of the tasks varied from 30s to 90s. The data was recorded using an Intel RealSense D415 with 30Hz. The OpenPose [21] was used to extract human pose as shown in Fig. 1. We compared our method to several state-of-art methods, including FCNN, FCNN with RLS-PAA (NN-RLS) [20], ERD [7], and LSTM-3LR [7]. The data used in the experiment is available at github.com/intelligent-control-lab/Human_Assembly_Data.

The proposed RNNIK-MKF has one recurrent layer with 128 hidden size. To establish a fair comparison, we designed the FCNN with two ReLU layers and a linear output layer [19]. We decreased the hidden size of the ERD and LSTM-3LR to match the size of ours. All models have the loss function set to be the Huber loss [22] and the post-training loss for all models was around 5×10^{-3}. All models were trained using the trajectories randomly selected from humans 1 and 3 doing tasks 1, 2, 3. Since the regular human reaction time to visual stimulus is around 0.25s, we set the input horizon be 10, which is equivalent to 0.3s. The experiments tested the prediction horizon from 1 to 60, which is equivalent to

Algorithm 2 RNNIK-MKF Motion Prediction

```
Input: Pre-trained RNN, Arm Model, $\phi_0 = \phi_{trained}, A = I$.
Input: MKF parameters: $0 < \lambda^w \leq 1$, $\sigma_w^w \geq 0$, $\sigma_v^w \geq 0$, $0 < \lambda^a \leq 1$, $\sigma_w^a > 0$, $\sigma_v^a \geq 0$.
Input: EMA parameters: $\mu_v^w,\ \mu_p^w,\ \mu_v^a,\ \mu_p^a \in [0, 1)$.
Output: Full arm trajectory prediction $\hat{y}$
for $i = 1, 2, \dots, k$ do
    Obtain current configuration $w_i$, $e_i$.
    $x_i^w = [w_{i-N+1}; \dots; w_{i-1}; w_i]$, $w_{pre} = w_i$.
    for $j = 1, 2, \dots, M$ do
        if $j = 1$ then
            if Observation Available then
                $\phi_i = MKF(\phi_{i-1},\ w_i - \hat{w}_1^{i-1}, h_{i-1})$
                $X_{i-1}^a = J^\mathsf{T}(\theta_{i-1})(w_i - w_{i-1})$
                $A_i = MKF(A_{i-1},\ \theta_i - \hat{\theta}_1^{i-1},\ X_{i-1}^a)$
                $\theta_{pre} = IK_{ew}([w_i;\ e_i])$
            else
                $\phi_i = \phi_{i-1}, A_i = A_{i-1}$
                $\theta_{pre} = IK_{ew}([w_i;\ \hat{e}_1^{i-1}])$
            end if
        end if
        $\hat{w}_j^i = RNN(\phi_i, x_i^w)$
        $x_i^w = [w_{i-N+j}; \dots; w_i; \dots; \hat{w}_j^i]$
        $\hat{\theta}_j^i = \theta_{pre} + A_i J(\theta_{pre})^\mathsf{T}(\hat{w}_j^i - w_{pre})$
        $[w_{pre}; \hat{e}_j^i] = FK_{ew}(\hat{\theta}_j^i)$, $\theta_{pre} = \hat{\theta}_j^i$
    end for
    $\hat{y}_i = [\hat{w}_1^i; \hat{e}_1^i; \dots; \hat{w}_M^i; \hat{e}_M^i]$
end for
```

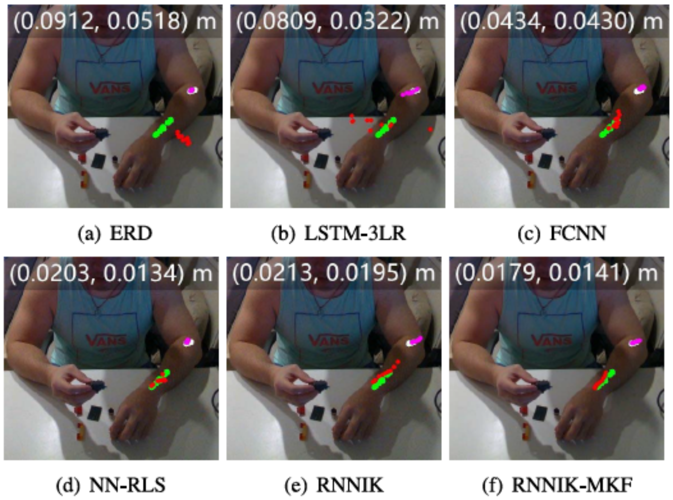
0.033 s to 2 s. We define the average prediction error for each prediction step as, $E_j = \frac{1}{T}\sum_{i=0}^{T}(||p_{i+j} - \hat{p}_j^i||_2)$, to quantitatively evaluate the prediction accuracy. T denotes the total time steps. In the following plots, RNNIK represents the proposed method with wrist MKF turned off.

A. Prediction Experiments

We used trajectories from humans 1 and 3 doing tasks 1, 2, 3 but different trials to evaluate the prediction quality. Fig. 3 shows the predicted trajectories from different methods at one time instance, where RNNIK-MKF outperforms others in terms of prediction accuracy. Figure 4 demonstrates the overall prediction accuracy quantitatively. As shown in Fig. 4, the proposed RNNIK-MKF and RNNIK had very close performance, which was shown in Fig. 3 as well. In general, the RNNIK-MKF had the lowest prediction error. It had 14% lower prediction errors on average comparing to LSTM-3LR, which had the lowest errors among the comparing methods. Fig. 5 shows the ratio of prediction error over the motion range. When predicting within 1*s*, RNNIK-MKF can maintain the error at around 10% relative to the motion range. For longer prediction step, the percentage can be maintained at around 15%. From the experiments, we can see that the proposed RNNIK-MKF can generate high-fidelity human arm motion predictions that are competitive to state-of-art methods.

B. Unseen Humans Experiments

We tested the methods with trajectories from human 2 doing tasks 1, 2, 3, to verify the effect of MKF. Figure 6 indicates that NN-RLS and RNNIK-MKF had significantly smaller prediction errors. Methods without adaptation had large errors and did not achieve similar performance as in the trained environment. In terms of adaptation, the RLS-PAA effectively reduced the error generated in FCNN when the prediction step was less than 20. But it exploded for longer prediction steps. On the other hand, RNNIK-MKF stably reduced the error comparing to RNNIK. It kept the maximum prediction error under 4 cm, which was at least 70% lower than other methods without adaptation. Since the data contains humans with



(a) ERD (b) LSTM-3LR (c) FCNN
(d) NN-RLS (e) RNNIK (f) RNNIK-MKF

Fig. 3. Visualizations of the predicted trajectories (10 steps) using different methods. Green: wrist ground-truth trajectories. White: elbow ground-truth trajectories. Red: wrist predictions. Purple: elbow predictions. Due to projection, the errors might appear to be larger than the true errors. True errors are labeled as (Wrist Error, Elbow Error) m.

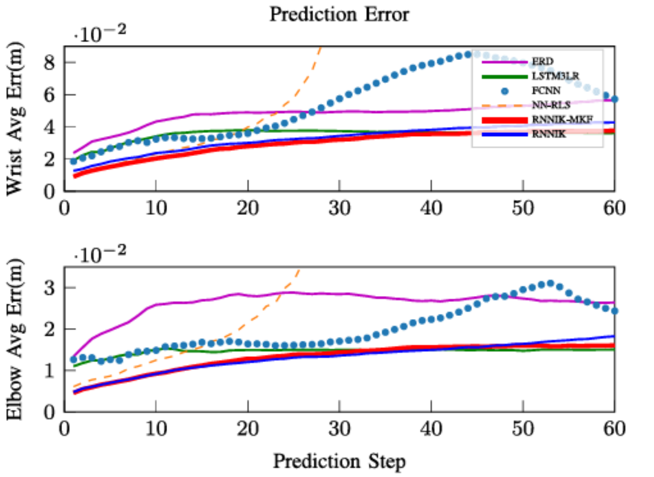


Fig. 4. Prediction Error for 1-60 Prediction Steps. Training and test data are from the same person doing the same tasks but different trials.

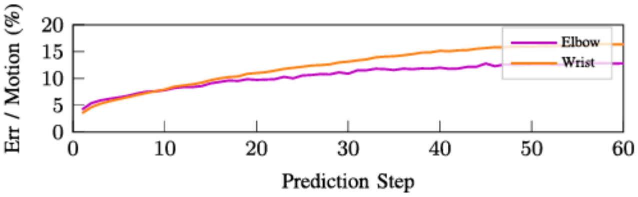


Fig. 5. Prediction Error Relative to Motion Range.

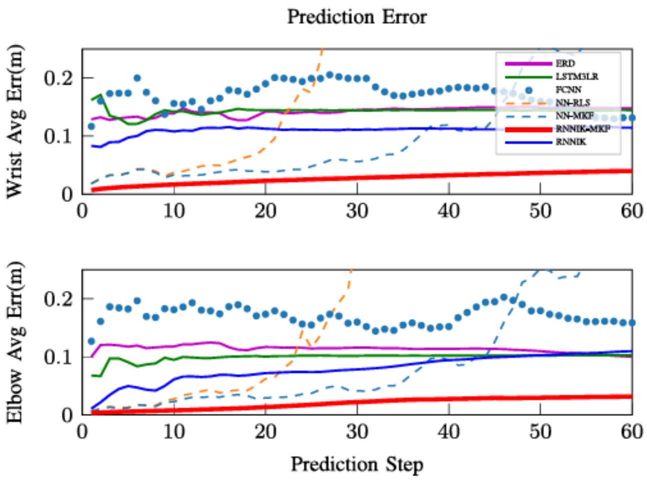


Fig. 6. Prediction Error for 1-60 Prediction Step. Training and test data are from different people doing the same tasks.

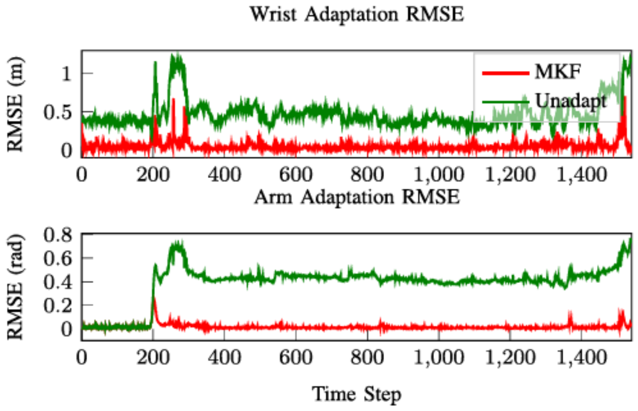


Fig. 7. RMSE of the error for online adaptation.

different heights and arm lengths, the results demonstrate that the method is robust across different scales of motion.

To directly compare the performance of MKF and RLS-PAA, we applied the MKF to FCNN. Figure 6 demonstrates that NN-RLS and NN-MKF performed similarly for short prediction horizon. NN-MKF was able to maintain the error stable for a longer prediction. However, it still exploded for predicting more than 35 steps. Thus, we can conclude that the proposed MKF is more robust than RLS-PAA for online adaptation since it has a virtual noise model and internal smoothing techniques. In addition, RNNIK outperformed FCNN since NN-MKF was still unstable for longer prediction comparing to RNNIK-MKF.

In addition, we investigated the adaptation error of MKF. Figure 7 shows the root mean square error (RMSE) for wrist and arm predictor with and without MKF. There existed a peak at time 200 for both, which was caused by sudden direction change by the human. But the peak was much smaller when MKF was turned on. There existed multiple smaller spikes for wrist. This was because we didn't encode the context (e.g., task state), while the wrist motion heavily depends on the context. On the other hand, there were fewer spikes for the elbow, since the behavior remained relatively stable throughout the task. The plot demonstrates that MKF effectively reduced the error caused by model mismatch. Figure 7 also indicates a fast convergence by MKF. The RMSE for arm settled within 100 steps, which was equal to 3.3 s. The RMSE for wrist settled faster within 20 steps, which was around 0.67 s. The settling times were different because the wrist adaptation started with a parameter matrix $\phi = \phi_{trained}$, which had prior knowledge encoded, while the arm adaptation started with $A = I$.

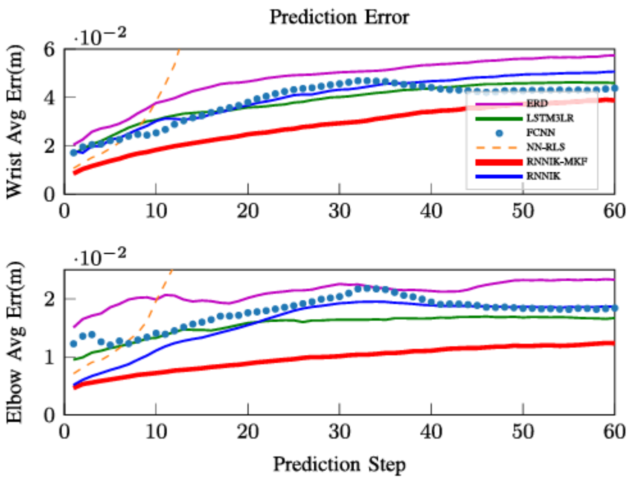


Fig. 8. Prediction Error for 1-60 Prediction Step. Training and test data are from the same person doing different tasks.

C. Unseen Tasks Experiments

We also tested the online adaptation using trajectories from humans 1 and 3 doing tasks 4 and 5. Figure 8 shows that the proposed RNNIK-MKF achieved the lowest prediction errors among all. The tasks in training and testing have different high-level features, but likely to share similar low-level features, since they were done by the same humans. From the results in Fig. 8, the adaptation successfully adapted the network to the new tasks, hence, made the model generalize to new tasks.

D. Motion Occlusion Experiments

We used the same data from Section IV-A to test the situation with occlusion. Several random segments were blocked to the pipeline. We mainly consider the occlusion on the elbow joint. When occlusion happens, the framework assumes that the observation aligns with the prediction and turns off the adaptation as shown in Algorithm 2. Hence, the occlusion problem is essentially equivalent to a longer-term prediction problem under the N-to-1 structure. For instance, when at time k, the second step predictions, $\hat{w}_2^k$ and $\hat{e}_2^k$, are predicted with input being $[w_{k-N+2}; e_{k-N+2}; \cdots; w_k; e_k; \hat{w}_1^k; \hat{e}_1^k]$. This is equivalent to when occlusion happens at time $k+1$. The first step predictions at time $k+1$, $\hat{w}_1^{k+1}$ and $\hat{e}_1^{k+1}$, are predicted with input being $[w_{k-N+2}; e_{k-N+2}; \cdots; w_k; e_k; \hat{w}_1^k; \hat{e}_1^k]$.

Therefore, we only considered the error for the first step prediction and we used the error metric $E = \frac{1}{T}\sum_i^T(||\hat{o}_1^i - \hat{p}_1^i||_2)$ to describe the variation between the predictions. T denotes the total steps of occlusion and $\hat{o}_1^i$ denotes the prediction with occlusion. The algorithm is considered robust if the variation is small between the predictions with and without occlusion.

Table I shows the variation for the wrist and elbow predictions of each algorithm. ERD and LSTM-3LR had significantly larger variations for wrist predictions than elbow

TABLE I
PREDICTION VARIATION WITH OCCLUSION

Algorithm	Wrist Variation (m)	Elbow Variation (m)
ERD	0.05537	0.02591
LSTM-3LR	0.03059	0.01468
FCNN	0.01795	0.01619
RNNIK	**0.0**	0.03042
RNNIK-MKF	**0.0**	**0.01164**

predictions although the occluded joint was elbow. This is because the models coupled the wrist and elbow in prediction. The elbow is determined by the upper three joints while the wrist is affected by all five joints. Thus, the wrist predictions had larger accumulated variation. On the other hand, since RNNIK and RNNIK-MKF decoupled the predictions with arm model, the occlusion of the intermediate arm would not influence the prediction of the wrist. However, RNNIK had the largest elbow variation while RNNIK-MKF had the lowest elbow variation, which was 20% lower than others. This demonstrated that MKF successfully reduced the error by mismatched models. From Table I we can infer that the proposed RNNIK-MKF is more robust compared to the existing methods when the arm motion is partially blocked from view.

V. CONCLUSION

This letter proposed a novel RNNIK-MKF human arm motion prediction framework. The method uses the RNN to predict the wrist motion and IK to extend the wrist prediction to full-arm prediction based on the physical arm model. The proposed MKF adapts the model in real-time to the current user or task. By comparing to several state-of-art methods, our method outperforms by showing that it can predict the arm motion with 14% lower prediction errors; it is generic to unseen humans or tasks since it has 70% lower prediction errors; it is more robust when partial arm motion is blocked as the occlusion has no impact on wrist prediction and has 20% less influence on elbow prediction.

The proposed RNNIK-MKF motion prediction framework can also be extended to robot-robot collaboration for predicting the motion of the co-robot. The prediction can help the ego robot to interpret the control policy of the other robots, and thus, collaborate more efficiently and safely. In the future, the wrist prediction could be extended to embed contextual information as mentioned in Section IV-B for potential better prediction.

REFERENCES

[1] P. A. Lasota, T. Song, and J. A. Shah, "A survey of methods for safe human–robot interaction," *Found. Trends Robot. Series*, vol. 5, no. 4, pp. 261–349, 2017.

[2] H. Ding, G. Reissig, K. Wijaya, D. Bortot, K. Bengler, and O. Stursberg, "Human arm motion modeling and long-term prediction for safe and efficient human–robot-interaction," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 5875–5880.

[3] J. Mainprice, R. Hayne, and D. Berenson, "Predicting human reaching motion in collaborative tasks using inverse optimal control and iterative re-planning," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2015, pp. 885–892.

[4] C. T. Landi, Y. Cheng, F. Ferraguti, M. Bonfè, C. Secchi, and M. Tomizuka, "Prediction of human arm target for robot reaching movements," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2019, pp. 5950–5957.

[5] S. Arai, A. L. Pettersson, and K. Hashimoto, "Fast prediction of a worker'S reaching motion without a skeleton model (F-PREMO)," *IEEE Access*, vol. 8, pp. 90340–90350, 2020.

[6] H. Salehinejad, S. Sankar, J. Barfett, E. Colak, S. and Valaee, "Recent advances in recurrent neural networks," 2018. [Online]. Available: arXiv:1801.01078.

[7] K. Fragkiadaki, S. Levine, P. Felsen, and J. Malik, "Recurrent network models for human dynamics," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2015, pp. 4346–4354.

[8] J. Martinez, M. J. Black, and J. Romero, "On human motion prediction using recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 4674–4683.

[9] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-rnn: Deep learning on spatio-temporal graphs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 5308–5317.

[10] Y. Cai *et al.*, "Learning progressive joint propagation for human motion prediction," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 226–242.

[11] B. D. Ziebart *et al.*, "Planning-based prediction for pedestrians," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2009, pp. 3931–3936.

[12] A. Rudenko, L. Palmieri, and K. O. Arras, "Joint long-term prediction of human motion using a planning-based social force approach," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2018, pp. 4571–4577.

[13] A. Akhtar, L. J. Hargrove, and T. Bretl, "Prediction of distal arm joint angles from emg and shoulder orientation for prosthesis control," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2012, pp. 4160–4163.

[14] L. Qiu *et al.*, "Peeking into occluded joints: A novel framework for crowd pose estimation," in *Proc. ECCV*, 2020, pp. 488–504.

[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[16] C. Welman, "Inverse kinematics and geometric constraints for articulated figure manipulation [microform]," M.S. thesis, School of Comput. Sci., Can. Theses Microfiche, Simon Fraser Univ., Burnaby, BC, Canada, 1993.

[17] J. Kiefer and J. Wolfowitz, "Stochastic estimation of the maximum of a regression function," *Ann. Math. Statist.*, vol. 23, no. 3, pp. 462–466, 1952.

[18] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control* (Dover Books on Electrical Engineering). Mineola, NY, USA: Dover Publ., 2014. [Online]. Available: https://www.google.com/books/edition/Adaptive_Filtering_Prediction_and_Contro/0_m9j_YM91EC?hl=en&gbpv=0

[19] A. Abuduweili and C. Liu, "Robust nonlinear adaptation algorithms for multitask prediction networks," *Int. J. Adapt. Control Signal Process.*, to be published.

[20] Y. Cheng, W. Zhao, C. Liu, and M. Tomizuka, "Human motion prediction using semi-adaptable neural networks," in *Proc. Amer. Control Conf. (ACC)*, 2019, pp. 4884–4890.

[21] Z. Cao, G. Hidalgo, T. Simon, S. Wei, and Y. A. Sheikh, "Openpose: Realtime multi-person 2D pose estimation using part affinity fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 172–186, Jul. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8765346

[22] P. J. Huber, "Robust estimation of a location parameter," *Ann. Math. Stat.*, vol. 35, no. 1, pp. 73–101, 1964.