MUSCLES KNOW FIRST: DECODING HAND MOTION FROM WRIST EMG FOR LATENCY-COMPENSATED VISION-BASED ROBOT TELEOPERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-based teleoperation has become the most accessible way to drive a dexterous robot hand, but a dramatic delay (150–400 ms) of robot hand motion against the operator's current pose forces the operator to slow down and wait for the robot, owing to the inherent flaws of the vision-based method. To overcome this limitation, we present Muscles Know First (MKF), a latency-conditioned two-stream forecaster that fuses the fresh surface electromyography (sEMG) stream with the delayed pose stream, predicting the operator's pose passed onto the robot. Experiments are separately conducted with motion tracking and hand–object contact tasks. For the motion tracking task, MKF recovers 61% of the tracking error caused by latency on unseen users with non-accumulated characteristic, and three control groups attribute the sEMG contribution to its timing rather than to its content. In a MuJoCo Shadow Hand testbed with three contact tasks timed to the operator's motion, the proposed MKF restores success rate from 57/26/13% to 92/83/67%, respectively, under 300 ms of latency with an inference time of only 8.5 ms. The outcomes of this study reveal a promising way for low-latency and more accurate robot hand teleoperation, potentially promoting dexterous robotic manipulation.

1 INTRODUCTION

Teleoperation is one of the principal pipelines through which robots acquire manipulation data (Khazatsky et al., 2024; Zhao et al., 2023). Its vision-based form, in which an RGB camera observes the operator's bare hand (Qin et al., 2023), is the most accessible, since it requires no glove, exoskeleton or markers. The camera channel, however, is slow. Markerless hand tracking on commercial headsets lags the hand by 45 ms on average (Abdlkarim et al., 2024) and by more than 220 ms on average at the onset of motion (Godden et al., 2025). Transport and synchronization raise the latency of the pose stream to 50–300 ms (Appendix F), and robot-side actuation adds about 100 ms (Kang & Hong, 2025; Liu et al., 2026). The robot therefore executes a pose that the operator has already left. Operators adopt the move-and-wait strategy (Sheridan & Ferrell, 1963; Ferrell, 1965), the robot teleoperation system AnyTeleop instructs them to slow down because hand tracking fails during fast motion (Qin et al., 2023), and in simulated vision-based vehicle teleoperation 150 ms of added latency halves a lane-keeping controller's route completion (Khalil & Kwon, 2026b).

Existing remedies leave this gap open. Vision-based systems make individual stages fast (Qin et al., 2023; Yin et al., 2025), and the remaining delay is executed unchanged. Prediction is the classical response to delay (LaValle et al., 2014; Smith & Jensfelt, 2010), yet motion history only records decisions that the motor system has already made, and extrapolating it fails when the hand starts, stops or reverses. Off-the-shelf generative prediction of the delayed image is not yet both accurate and fast (Khalil & Kwon, 2026a), and discrete-goal prediction (Kim et al., 2026a) does not address the continuous pose stream. Surface electromyography (sEMG) has been used to anticipate motion and compensate delay (Feleke et al., 2021; Kang & Hong, 2025; Kang et al., 2026), but only for arm and wrist motion, for delays fixed in advance, and with models trained and tested on the same

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

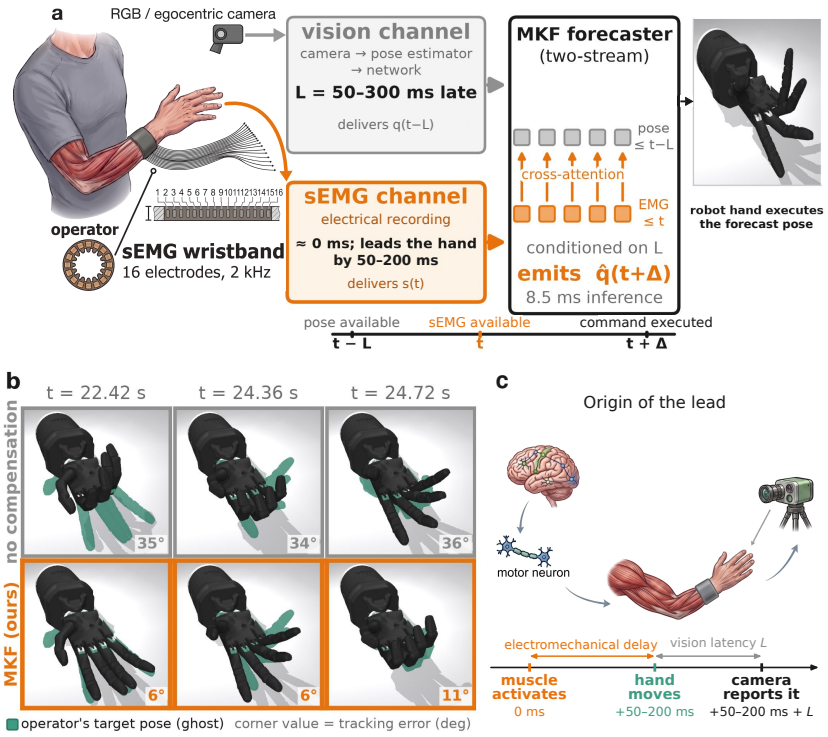

Figure 1: **(a)** The vision pipeline delivers the operator's pose $\mathbf{q}(t-L)$ after $L = 50$–300 ms, while an sEMG wristband delivers the muscle state $\mathbf{s}(t)$ with negligible delay, 50–200 ms before the hand moves. MKF fuses the two streams into $\hat{\mathbf{q}}(t+\Delta)$, the pose the robot should hold when its command takes effect. **(b)** A held-out user's recording streamed through a simulated Shadow Hand at L=300 ms; teal ghost: the operator's true pose; corner values: tracking error (deg). The three frames are the moments of largest advantage in a fast-motion window (Appendix B). **(c)** The motor command reaches the muscle before the fingers move, and the camera reports the motion a further L later.

person. Whether this anticipation extends to the twenty degrees of freedom of the hand, to operators unseen during training, and to a vision delay that is not known in advance has not been established.

There is a physiological reason to expect that it does. sEMG records the electrical activity that drives a muscle, and this activity precedes the resulting force and motion: at movement onset by an electromechanical delay of 50–100 ms (Feleke et al., 2021; Kang & Hong, 2025), and during continuously modulated contractions by a neuromechanical delay of about 200 ms between neural drive and force (70–385 ms; Del Vecchio et al., 2018). Since this delay is a property of muscle physiology, the lead may transfer across operators. sEMG is also acquired electrically, so when the vision pipeline delivers the pose at $t - L$, a wristband has already delivered the muscle state at t. Relative to a delayed camera, the EMG signal therefore leads by two terms, the electromechanical delay and the pipeline latency L (Fig. 1). In teleoperation the second term dominates, because the muscles have already begun a movement that the pose stream cannot yet show (Sec. 4).

We propose **Muscles Know First (MKF)**, a teleoperation framework that pairs the vision pipeline with an sEMG wristband and uses the wristband to compensate the latency of the pipeline. Its forecaster reads one second of hand pose ending at $t - L$ and one second of muscle activity ending at t, fuses them by cross-attention, and predicts the operator's pose up to 500 ms ahead, so that the robot executes the pose the operator will hold when the command takes effect. The latency L is an input, so a single model serves any vision delay from 0 to 300 ms, and the forecast is re-anchored to the newest pose at every control tick, so errors do not accumulate during continuous execution.

We train and evaluate on emg2pose (Salter et al., 2024), the largest public dataset of paired sEMG and hand pose (370 hours, 193 users, 16-channel 2 kHz sEMG with motion-capture labels), using its three official held-out splits. On unseen users, MKF recovers 61% of the tracking error that latency

adds. Three control groups, in which the EMG stream is shuffled in time, delayed by the same L as vision, or zeroed during training, attribute the gain to the timing of the EMG signal. Streamed tick by tick over 900 test files, MKF reduces tracking error relative to no compensation by 26.0–33.5% at L=100–300 ms without accumulating error, and the gain persists under tracking noise measured on three real egocentric chains. In a MuJoCo Shadow Hand testbed, MKF recovers 82%, 77% and 62% of the catches, gate clearances and timed releases lost to 300 ms of latency.

Our contributions are (i) the first large-scale, cross-user measurement of how much anticipatory information sEMG carries about future hand pose; (ii) MKF, a latency-conditioned two-stream forecaster that models the differential latency between the vision and EMG channels and serves $L \in [0, 300]$ ms with a single model; and (iii) a simulated teleoperation testbed that evaluates latency compensation in continuous hand tracking and in contact tasks timed to the operator's motion.

2 Related Work

Latency in teleoperation. Operators facing delay adopt a move-and-wait strategy (Sheridan & Ferrell, 1963; Ferrell, 1965), and prediction ahead of the delay has been the standard remedy: predictive displays (Bejczy et al., 1990), minimum-jerk prediction of the operator's hand input (Smith & Jensfelt, 2010), head-orientation prediction in virtual reality (LaValle et al., 2014), and body-motion forecasting for latency compensation (Bohnerth et al., 2025; Wei et al., 2025). On the robot side, real-time action chunking and its successors treat inference latency as an explicit quantity and evaluate with injected delays (Black et al., 2025; Wang et al., 2026; Kim et al., 2026b). All of these methods predict from the delayed signal itself; none uses a signal that leads the motion.

Dexterous teleoperation systems. AnyTeleop combines a hand detector with kinematic retargeting (Qin et al., 2023), Open TeleDex improves the synchronization of multiple sensor streams (Chi et al., 2025), and GeoRT performs geometric retargeting at 1 kHz (Yin et al., 2025). These systems improve individual stages of the pipeline.

sEMG for delay compensation and robot control. Muscle activity has been used to anticipate motion at the arm level and at small scale: Feleke et al. (2021) predict hand position from arm EMG up to 250 ms ahead for individual subjects, Kang & Hong (2025) train a dilated CNN with 38k parameters to predict the wrist 100 ms ahead, compensating a 100 ms motor delay, with one model per user (Kang et al., 2026), and Machida & Kameyama (2026) predict planar arm motion. These settings correspond to L=0 in ours. Decoding the *current* hand pose or gesture from sEMG is an active benchmark (Salter et al., 2024; Xi et al., 2026; Gilardini et al., 2026; Cui et al., 2026), supported by wrist-worn hardware (Kaifosh et al., 2025). DexEMG drives a 22-DoF hand from a forearm armband by integrating decoded joint velocities, with per-user calibration (Zhao et al., 2026), and EMG has served robots as a channel for intent or effort (DelPreto & Rus, 2019; Xiong et al., 2024; Bernardini et al., 2026; Liu et al., 2026) and, recently, as a modality for collecting demonstrations for policy learning (He et al., 2026). In this literature EMG describes the present state of the hand. We ask whether it also provides lead time that is preserved during execution.

Egocentric hand forecasting. Methods in this area predict future hand pose from images or video over horizons from about a third of a second to several seconds (Prakash et al., 2025; Hatano et al., 2025; Shi et al., 2026; Hoque et al., 2026), longer than the horizons below 300 ms considered here, over which the motor command has already been issued.

3 Problem Formulation and Method

Latency model. Let $\mathbf{q}(t) \in \mathbb{R}^{20}$ be the operator's joint-angle trajectory and $\mathbf{s}(t) \in \mathbb{R}^{16}$ the wristband signal. At wall-clock time t, a vision-based interface has observed only

$$\mathcal{O}_{\text{vis}}(t) = \{\mathbf{q}(\tau) : \tau \le t - L\}, \qquad \mathcal{O}_{\text{emg}}(t) = \{\mathbf{s}(\tau) : \tau \le t\}, \tag{1}$$

where L is the *differential* latency of the vision chain relative to the EMG chain, that is, capture, hand-pose inference and transport, minus the negligible electrical acquisition time of sEMG. Delays downstream of the interface (retargeting, network and actuation) are absorbed into a horizon Δ: a command issued at t is executed at $t + \Delta$, so the interface must output an estimate of $\mathbf{q}(t + \Delta)$. Deployed systems issue the newest available pose, $\mathbf{c}_{\text{zv}}(t) = \mathbf{q}(t - L)$, and therefore execute the

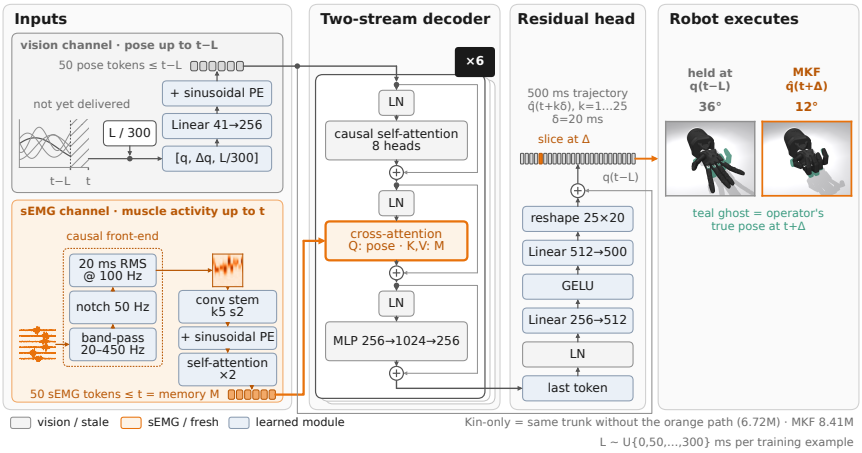

Figure 2: **The architecture of the MKF forecaster.** Pose tokens ending at $t-L$, together with L, attend through cross-attention to sEMG tokens ending at t, and a residual head outputs the next 500 ms as offsets from $\mathbf{q}(t-L)$. Right: an example window at L=300 ms; the values are the errors (deg) of the held pose and of the MKF output relative to the operator's pose at t+100 ms.

operator's motion $L+\Delta$ late. A latency-compensating interface issues

$$\mathbf{c}(t) = \hat{\mathbf{q}}(t+\Delta) = f_\theta\big(\mathcal{O}_{\text{vis}}(t),\, \mathcal{O}_{\text{emg}}(t),\, L\big), \tag{2}$$

a forecast over a horizon of $L+\Delta$ for the pose stream and of only Δ for the EMG stream, effectively less because muscle activation precedes force by the electromechanical delay. Training covers L from 0 to 300 ms, beyond the mean onset latency of 229 ms reported by Godden et al. (2025), and Δ up to 500 ms.

Asymmetric two-stream forecaster (Fig. 2). The forecaster takes N=50 pose frames at 50 Hz ending at $t-L$ and N EMG tokens ending at t. Pose tokens are $\mathbf{x}_i = \mathbf{W}_p[\mathbf{q}_i;\, \mathbf{q}_i - \mathbf{q}_{i-1};\, L/300] + \mathbf{p}_i$ with sinusoidal positional encodings $\mathbf{p}_i$. The EMG stream is causally filtered to a 100 Hz RMS envelope (Appendix A), embedded by a stride-2 convolutional stem with the same positional encodings, and encoded by two self-attention blocks into a memory $\mathbf{M} \in \mathbb{R}^{N\times d}$ (d=256). Each of the six trunk blocks computes, with pre-layer normalization and residual connections,

$$\mathbf{X} \leftarrow \mathbf{X} + \mathrm{SA}_{\text{causal}}(\mathbf{X}), \qquad \mathbf{X} \leftarrow \mathbf{X} + \mathrm{CA}(\mathbf{X}, \mathbf{M}), \qquad \mathbf{X} \leftarrow \mathbf{X} + \mathrm{MLP}(\mathbf{X}), \tag{3}$$

where $\mathrm{CA}(\mathbf{X},\mathbf{M}) = \mathrm{softmax}\big(\mathbf{X}\mathbf{W}_Q(\mathbf{M}\mathbf{W}_K)^\top/\sqrt{d_k}\big)\mathbf{M}\mathbf{W}_V$ with 8 heads, so that every pose query, whose newest content is L old, can attend to muscle activity up to the present. The head reads the last token $\mathbf{h} = \mathrm{LN}(\mathbf{X}_N)$ and predicts a trajectory as a residual to the stale pose (Martinez et al., 2017),

$$\hat{\mathbf{q}}(t+k\delta) = \mathbf{q}(t-L) + \big[\mathbf{W}_2\,\mathrm{GELU}(\mathbf{W}_1\mathbf{h})\big]_k, \qquad k=1,\dots,25, \quad \delta = 20\,\text{ms}, \tag{4}$$

so that one forward pass (8.5 ms; Table 10) yields the next 500 ms, and the robot reads the output at its own Δ. Since L is sampled independently for each training example, one network serves every latency. The objective is the validity-masked Huber loss over all outputs,

$$\mathcal{L}(\theta) = \mathbb{E}_{L\sim\mathcal{U}\{0,50,\dots,300\}\,\text{ms}}\,\mathbb{E}_t\Big[\, v_t \cdot \tfrac{1}{25\cdot 20}\textstyle\sum_{k,j}\rho\big(\hat{q}_j(t+k\delta) - q_j(t+k\delta)\big)\Big], \tag{5}$$

where ρ is the smooth-ℓ_1 penalty on standardized angles and $v_t \in \{0,1\}$ excludes windows that contain a motion-capture IK failure, following the official emg2pose protocol. We refer to the full model as **MKF** (8.41M parameters) and to its kinematics-only ablation as **Kin-only** (6.72M), which is the same network without the EMG path and cross-attention and with an 8-block trunk.

Baselines without training. Learned models are compared with classical predictors that extrapolate the stale pose over the full horizon $h = L+\Delta$: zero velocity $\hat{\mathbf{q}}_{\text{zv}} = \mathbf{q}(t-L)$ (Martinez et al., 2017), constant velocity $\hat{\mathbf{q}}_{\text{cv}} = \mathbf{q}(t-L) + \hat{\mathbf{v}}\,h$ and constant acceleration $\hat{\mathbf{q}}_{\text{ca}} = \hat{\mathbf{q}}_{\text{cv}} + \frac{1}{2}\hat{\mathbf{a}}\,h^2$, with $\hat{\mathbf{v}}, \hat{\mathbf{a}}$ estimated by per-joint Kalman filters (by one-frame finite differences in Sec. 5), and a minimum-jerk predictor following Smith & Jensfelt (2010), which in our implementation comes to rest 500 ms

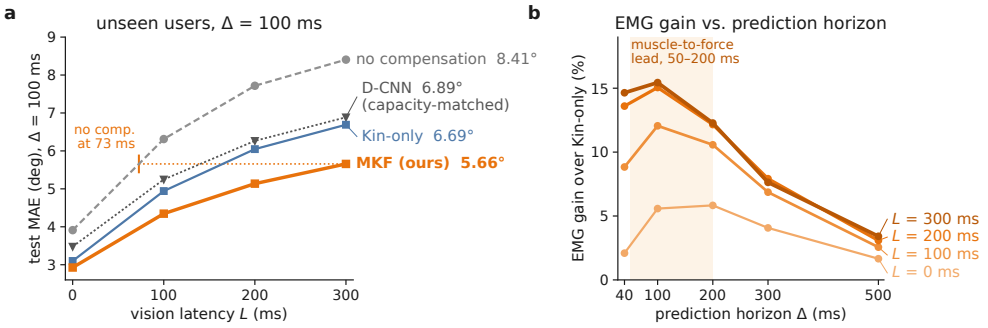

Figure 3: **Forecasting error versus vision latency, and EMG gain versus prediction horizon. (a)** Test MAE versus vision latency L (unseen users, Δ=100 ms; bands span two seeds). The dotted line marks the error-equivalent latency of MKF at L=300 ms. **(b)** Gain of MKF over Kin-only versus prediction horizon Δ, one curve per L.

Table 1: **Latency-conditioned forecasting on emg2pose** (test, unseen users, Δ=100 ms). Joint-angle MAE (deg) and official-FK fingertip error (mm). The Δ-only rule extrapolates a 5-frame mean velocity over Δ only, and both D-CNN variants (Kang & Hong, 2025) receive MKF's inputs. Kin-only and MKF are two-seed means; EMG gain: $(\mathrm{Kin}-\mathrm{MKF})/\mathrm{Kin}$, mean±half-range.

	$L=0$		$L=100$ ms		$L=200$ ms		$L=300$ ms	
Method	deg	mm	deg	mm	deg	mm	deg	mm
No compensation (stale pose)	3.91	8.8	6.31	16.8	7.72	19.6	8.41	21.3
Const.-velocity (Δ only)	5.17	–	7.67	–	9.33	–	10.20	–
Const.-velocity, Kalman (L+Δ)	5.27	–	10.43	–	15.25	–	19.54	–
Const.-acceleration, Kalman (L+Δ)	8.23	–	23.72	–	46.61	–	76.74	–
Minimum-jerk (L+Δ)	5.64	–	11.18	–	14.73	–	16.23	–
D-CNN, 0.05M (Kang & Hong, 2025)	4.27	–	6.06	–	7.16	–	7.77	–
D-CNN, capacity-matched 7.7M	3.48	–	5.25	–	6.26	–	6.89	–
Kin-only: kinematics-only ablation, 6.7M	3.10	6.6	4.94	12.6	6.05	14.9	6.69	16.5
$\mathrm{MKF}_{\varnothing}$: MKF w/ EMG zeroed, 8.4M	3.11	–	4.96	–	6.09	–	6.73	–
MKF: kinematics+EMG (ours), 8.4M	**2.93**	**6.1**	**4.35**	**10.6**	**5.14**	**12.0**	**5.66**	**13.6**
EMG gain (MKF vs. Kin-only, 2 seeds)	$5.6 \pm 0.1\%$		$12.1 \pm 0.3\%$		$15.1 \pm 0.5\%$		$15.4 \pm 0.4\%$	
Latency-induced error recovered, Kin / **MKF**	–		57% / **82%**		44% / **68%**		38% / **61%**	

after $t - L$. Zero velocity, which holds the newest available pose, corresponds to deployed practice and is referred to as *no compensation*. Kin-only reads the clean motion-capture pose, an optimistic proxy for the optical channel; Sec. 5 repeats the comparison with the pose stream degraded to the error statistics of real egocentric tracking chains (Table 4).

Evaluation protocol and metrics. We follow the official emg2pose splits along the three generalization axes (held-out users, held-out gestures or stages, and both), score only frames with a valid motion-capture label, and report the joint-angle MAE, $\mathrm{MAE}_{\Delta} = \mathbb{E}\,\|\hat{\mathbf{q}}(t+\Delta) - \mathbf{q}(t+\Delta)\|_1/20$ in degrees, and the fingertip error in millimetres computed with the official forward kinematics. Main tables use Δ=100 ms, and headline results are means over two seeds with half-range spread. Two derived quantities summarize a latency sweep: the *error-equivalent latency* $L_{\mathrm{eq}}(L) = \min\{L' : \mathrm{MAE}_{\mathrm{zv}}(L') \geq \mathrm{MAE}_{\mathrm{MKF}}(L)\}$, at which the uncompensated system would match the compensated one, and, for tasks with binary outcomes, the *recovery* of successes lost to latency, $R = (S_{\mathrm{MKF}} - S_{\mathrm{zv}})/(S_{\mathrm{op}} - S_{\mathrm{zv}})$, where S_{op} counts the trials in which the operator's own zero-latency recording succeeds and S_{zv} those without compensation.

4 How Much Anticipatory Information Does EMG Carry?

Error under latency. Without compensation, at Δ=100 ms the error grows from 3.91° at L=0 to 8.41° at L=300 ms (Fig. 3a), an increase of 4.5° that we refer to as the latency-induced error. Kinematic forecasting (Kin-only) recovers 1.7° of this error and EMG (MKF) a further 1.0°. **In**

Table 2: **EMG-only command streams versus latency-compensated vision** (emg2pose test split, unseen users, MAE in deg). The released emg2pose checkpoints (tracking: true pose every 5.9 s; regression: no pose) are scored against the operator's pose Δ later; Δ=0 is their native score. EMGFormer: emg2pose user-split values from Xi et al. (2026). Vision rows: models of Table 1. n.d.: not defined.

Command stream	Δ=0	Δ=100 ms	Δ=300 ms
EMG only (no vision)			
vemg2pose, tracking	8.51	9.28	11.13
emg2pose, tracking	11.80	12.27	13.55
vemg2pose, regression	12.72	13.29	14.90
emg2pose, regression	12.96	13.55	15.18
NeuroPose, regression	13.57	14.00	15.13
EMGFormer-S/M/L, reported	12.45/12.40/12.26	n.d.	n.d.
EMG-only forecaster (ours, 6.8M)	n.d.	13.46	13.87
Vision (stale pose), with and without compensation			
No compensation, L=300 ms	n.d.	8.41	8.99
Kin-only, L=300 ms	n.d.	6.69	7.35
MKF, L=300 ms (ours)	n.d.	**5.66**	**6.79**

total, compensation recovers 61% of the latency-induced error (last row of Table 1), and EMG accounts for 38% of the recovered error. The gain of MKF over Kin-only is 5.6%, 12.1%, 15.1% and 15.4% at L=0, 100, 200 and 300 ms, and it holds on the harder axes, with 16.6% for unseen gestures and 12.5% for unseen users performing unseen gestures at L=300 ms (Table 12). **MKF at L=300 ms is more accurate than the uncompensated system at L=100 ms** (5.66° vs. 6.31°; error-equivalent latency 73 ms). At L=300 ms, every classical predictor that extrapolates over $L+\Delta$ is less accurate than holding the stale pose. The D-CNN EMG predictor of Kang & Hong (2025), scaled to 7.7M parameters and given the same inputs as MKF, performs close to Kin-only (6.89° vs. 6.69°).

Control experiments. Three control groups identify the source of the gain. *Time shuffle*: randomly reordering the EMG frames, which keeps their values and destroys their timing, removes the gain. At L=100 ms the shuffled model matches a Kin-only model trained with the same recipe (5.00° vs. 5.02°, +0.3%), whereas the intact MKF reaches 4.52° (+9.9%; Appendix D). *Synchronized delay*: if the value of EMG lay in its content alone, delaying it by the same L as vision would preserve the gain. For the model trained on fresh EMG, the gain falls from +15.6% with fresh EMG to +6.8% with EMG delayed by 100 ms and to −4.5% when EMG is as old as the pose; a model trained with EMG always as old as the pose retains +4.7%, +3.4% and +2.3% at L=100, 200 and 300 ms (Table 7). The content of delayed EMG is therefore worth a few percent, and its freshness accounts for the remaining seven to thirteen points. *Zeroed EMG*: MKF$_\varnothing$, the MKF architecture trained with the EMG stream set to zero, performs like Kin-only (within 0.04° at every L), so the additional capacity does not explain the gain; with the EMG input restored, the error decreases by 12.5–15.9% at $L \geq 100$ ms. EMG alone is insufficient: as EMG-only command streams, the released emg2pose decoders reach 13.3–14.0° at Δ=100 ms (Table 2), similar to our EMG-only forecaster (13.5°) and above the 8.41° of holding a 300 ms-old pose. At L=0, where only physiology separates the two streams, the gain is +5.6%, +5.8% and +4.1% at Δ=100, 200 and 300 ms, consistent with the lead of neural drive over force; the increase to +15.4% at L=300 ms reflects the difference in freshness. In Fig. 3b, the gain peaks at Δ=100 ms and falls to +3.4% at 500 ms.

Motion states and generalization across users. Under latency, a stale pose is least informative at transitions from rest: at L=300 ms MKF reduces the error on onset segments from 8.09° to 5.74° (Kin-only 7.28°), with the same ordering on steady motion and reversals. Before motion begins, EMG does not detect the upcoming movement better than kinematics alone (Appendix E). The gain saturates at about 40 training users, is nearly as large on unseen users as within subjects (Table 13), and does not improve with per-user fine-tuning (Appendix A), consistent with a lead shared across users.

Sensitivity to the latency estimate. MKF takes L as an input, which a deployed system must estimate. Table 3 covers three cases: **(A)** frames carry timestamps, so the latency of every frame is known exactly; **(B)** latency varies from frame to frame and only its nominal value is known; **(C)** latency is constant, but its measured value is off by a fixed amount. A calibration error of ±50 ms

Table 3: **Sensitivity to the latency estimate.** Unseen-user MAE (deg) of the seed-0 models at Δ=100 ms. The pose stream is delayed by the true L, and each model is told the value in the second column. The jitter rows pool the L=100, 200 and 300 ms runs.

Case	Model and what it is told about L	$L = 100$	$L = 200$	$L = 300$
ref.	No compensation (holds the stale pose)	6.31	7.72	8.41
ref.	Kin-only, told the true L	4.95	6.07	6.70
ref.	**MKF, told the true L**	**4.34**	**5.12**	**5.64**
C	MKF, told $L \pm 50$ ms (worst of $\pm 25, \pm 50$)	4.45	5.19	5.66
C	MKF, told $L \pm 100$ ms (worst)	4.76	5.31	5.75
C	MKF, told L=0 (no estimate at all)	4.76	6.40	7.50
Latency jumps per tick among $\{100, 200, 300\}$ *ms (nominal 200)*				
–	No compensation		7.47	
A	Kin-only, timestamped: told the true L each tick		5.90	
A	**MKF, timestamped: told the true L each tick**		**5.03**	
B	**MKF, told the constant nominal 200 ms**		**5.16**	

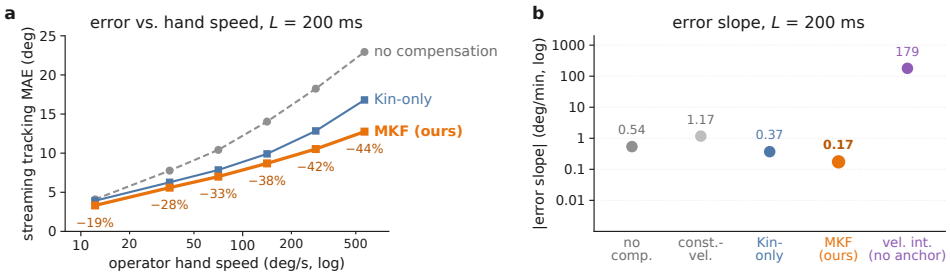

Figure 4: **Tracking error versus hand speed, and error slope over time** (continuous hand tracking over the 900 test files, L=200 ms). **(a)** Streaming MAE of each condition with ticks binned by true hand speed; the labels give the change in error of MKF relative to no compensation. **(b)** Error slope over 40 s files; vel. int.: MKF increments integrated without per-tick re-anchoring.

increases the error by at most 0.11° (2.6%) and one of $\pm$100 ms by at most 0.42° (9.8%), and every miscalibrated MKF remains 4–16% more accurate than Kin-only given the true L. Even without any estimate (MKF told L=0), the error stays 11–25% below no compensation. When latency jumps between 100 and 300 ms from tick to tick, timestamps (A) give 5.03° and a constant nominal value of 200 ms (B) gives 5.16°, which differ by 2.6%; (B) remains 31% below no compensation.

5 Executing the Forecasts: A Simulated Teleoperation Testbed

5.1 Continuous Hand Tracking

Section 4 evaluates windows independently. To test continuous execution, we stream a stratified 900-file sample of the official 6,167-file test split ($\sim$12 h) through a simulated hand that has no task or object and holds the commanded pose, so any difference from the operator's hand is due to the interface. At every 20 ms tick the model predicts again and the hand executes the forecast at Δ=20 ms. We report the per-tick MAE, the phase lag estimated by cross-correlation, and the error slope over each 40 s file.

Tracking error, phase lag and drift. Compared with no compensation, MKF reduces the per-tick tracking error by 26.0%, 31.6% and 33.5% at L=100/200/300 ms (Table 6). The cross-correlation lag over whole files changes little (307 vs. 320 ms at L=300 ms), since the reduction is concentrated in fast segments of motion (Fig. 4a). The error of constant-velocity extrapolation, the simplest remedy, is 2.4$\times$ that of no compensation at L=300 ms (18.0° vs. 7.66°), because its errors at motion reversals are executed at full amplitude. Integrating the increments predicted by MKF without re-anchoring, the reconstruction used by the velocity decoder of Zhao et al. (2026), drifts at 179°/min at L=200 ms, whereas re-anchored MKF drifts at 0.17°/min (Fig. 4b).

Table 4: **Latency compensation under vision degradation calibrated on real egocentric chains** (emg2pose test, unseen users, Δ=100 ms). Top: error of each tracking chain against motion-capture ground truth, as medians over 103 and 107 hand-clips of HOT3D (Banerjee et al., 2025) and 12 test subjects of EgoEMG (Xi et al., 2026): MAE and σ (deg), AR(1) time constant τ (ms), frame-to-frame jitter (deg), and rate (/s) and mean duration (ms) of tracking losses of at least 100 ms. Bottom: MAE (deg) of models retrained on the pose stream degraded to each chain.

Chain (camera, tracker)	Rate (Hz)	MAE	σ	τ	Jitter	Losses (/s; ms)
HOT3D Aria RGB, MediaPipe Hands (Zhang et al., 2020)	30	25.4	26.7	84	19.3	0.16; 670
HOT3D Aria RGB, WiLoR (Potamias et al., 2025)	30	12.3	8.9	148	4.6	0.22; 320
EgoEMG head camera, released estimator	60	6.7	7.9	515	2.1	0.0045; 2920

	HOT3D + MediaPipe				HOT3D + WiLoR				EgoEMG chain			
Method	L=0	100	200	300	L=0	100	200	300	L=0	100	200	300
No compensation	22.39	23.30	23.95	24.25	8.91	10.59	11.63	12.14	8.12	9.90	10.98	11.52
Kin-only	8.44	9.09	9.46	9.69	5.85	7.03	7.72	8.10	5.78	7.01	7.76	8.19
MKF (ours)	**7.98**	**8.38**	**8.63**	**8.85**	**5.56**	**6.41**	**6.89**	**7.21**	**5.56**	**6.45**	**6.96**	**7.31**
EMG gain	5.5%	7.8%	8.8%	8.7%	4.9%	8.8%	10.8%	11.0%	3.9%	8.1%	10.4%	10.8%

Error versus hand speed. Grouping the executed ticks by ground-truth hand speed (Fig. 4a) shows that the uncompensated error grows steeply with speed, while the relative error reduction of MKF increases monotonically from 19% for near-static ticks to 44% in the fastest band. Latency therefore penalizes fast motion most, which is the motion that AnyTeleop instructs operators to avoid (Qin et al., 2023), and compensation is most effective there.

Vision degradation calibrated on real egocentric chains. The evaluations above use the clean motion-capture pose as vision input. To account for real markerless tracking, we measured three egocentric tracking chains against independent motion-capture ground truth (Table 4, top; Appendix B.3): MediaPipe Hands, the tracker used by AnyTeleop, and WiLoR (Potamias et al., 2025), a recent 3D hand reconstruction model, both on the head-mounted RGB stream of HOT3D, and the head-camera chain of EgoEMG with its released in-domain estimator. Kin-only and MKF were retrained with the pose stream degraded to the frame rate, noise magnitude, correlation time and loss rate of each chain (Table 4, bottom). For all three chains the EMG gain is positive and increases with L, reaching $+8.7\%$, $+11.0\%$ and $+10.8\%$ at L=300 ms, compared with $+15.4\%$ on clean input. MKF therefore remains effective on the erroneous poses of real egocentric trackers: for all three chains, MKF at L=300 ms is more accurate than the tracked pose itself at L=0 (7.21° vs. 8.91° for WiLoR), so it also reduces part of the tracking error. During tracking-loss freezes, MKF continues to follow the operator from muscle activity alone (8.54° vs. 11.70° without compensation; Appendix D), and with 500 Hz servo dynamics the relative gains are unchanged (Fig. 8).

5.2 Contact tasks with deadlines

Tracking error does not fully capture the effect of latency on manipulation, in which many actions have deadlines. We therefore instrument the recorded motion with contact tasks timed by the operator's own motion (Fig. 5; constants in Table 11). Each task anchors a virtual object to an event on the operator's ground-truth stream, a closure or an opening onset, and every condition is replayed against the same schedule. Only events that the operator's own recording completes are scored (n=152, 47 and 30); the timing margins were fixed before any model was evaluated, and sensitivity analyses over the task parameters are given in Tables 14 and 15.

In the **catch** task, 300 ms of latency loses 43% of the operator-feasible catches (no compensation 57%). Constant-velocity extrapolation does not help (55%, p=0.804), Kin-only reaches 83% and MKF 92%, recovering 82% of the lost catches (exact McNemar test on paired events: $p=2\times10^{-15}$ vs. no compensation, p=0.009 vs. Kin-only). In the **gate** task, a late withdrawal results in a collision. At L=300 ms the uncompensated hand clears the plate in 26% of events; constant-velocity extrapolation helps in this task (55%) because withdrawal is a smooth motion, Kin-only reaches 60% and MKF 83% ($p=1\times10^{-7}$ vs. no compensation, $p=1\times10^{-3}$ vs. Kin-only), recovering 77% of the lost clearances. In the **timed release** task, the uncompensated hand releases too late in almost every event at L=300 ms (success 13%; 11 of 30 balls are never released, and the median delay is +500 ms). Kin-only reaches 40% and MKF 67% (median delay +60 ms; $p=4\times10^{-4}$ vs. no compensation, p=0.021 vs. Kin-only).

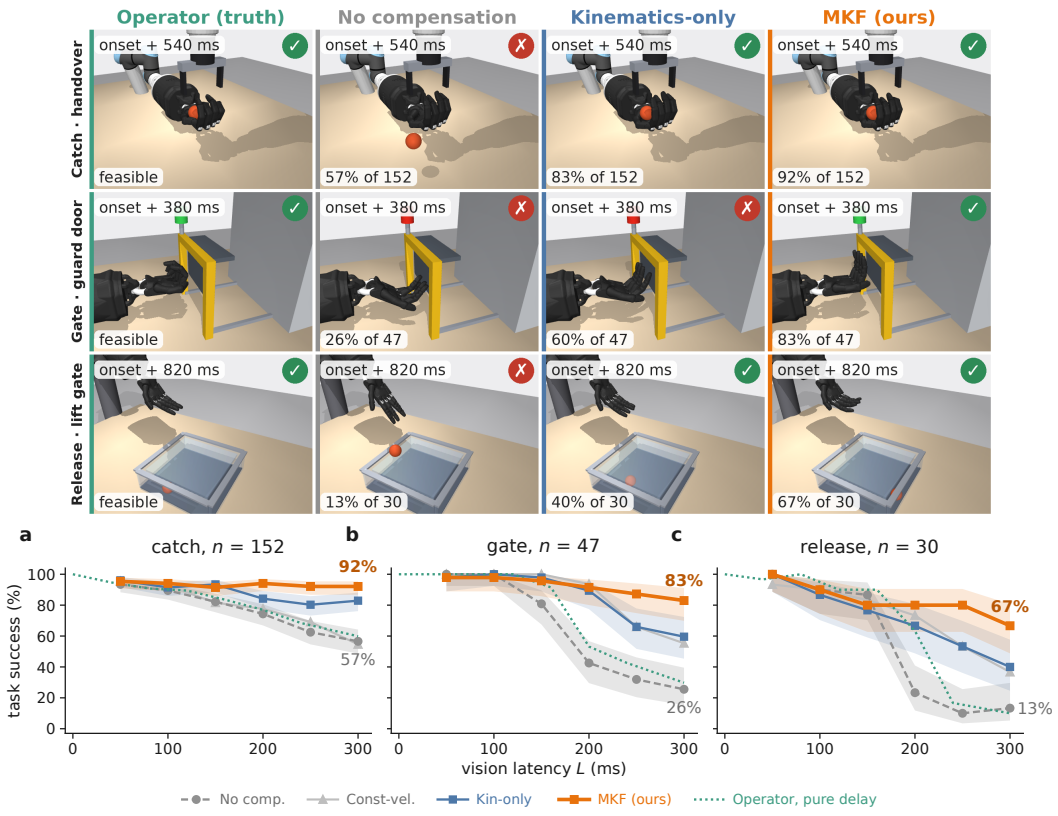

Figure 5: **Contact tasks timed to the operator's motion.** *Top:* one decisive frame per task and condition at L=300 ms, labelled with the time since the operator's event onset and the success rate over operator-feasible events (the wrist is fixed; arms, gripper and fixtures are visual context). **Catch**: a gripper releases a ball into the hand as the operator's closure begins. **Gate**: a door closes on the fingertips 200 ms after the operator withdraws them. **Release**: the ball must drop into a bin whose lift gate is open only within ±200 ms of the operator's release. *Bottom:* success versus vision latency (Wilson 95% CIs); dotted teal: the operator's own motion delayed by L.

Cost of compensation. At low latency, where little error remains to be recovered, a forecaster could perform worse than holding the delivered pose. At L=100 ms, the uncompensated hand succeeds in 89/100/90% of the catch, gate and release events, and MKF in 94/98/90%. At L=300 ms, MKF releases the ball prematurely in 7 of 30 release events, compared with 3 without compensation, because it predicts small motions while the operator holds still. Kin-only (7) and constant-velocity extrapolation (11) show the same failure, which indicates that it arises from forecasting during a static hold (Appendix C).

6 Conclusion

Inspired by motor physiology, we presented MKF, a latency-conditioned forecaster that combines the delayed pose stream of vision-based teleoperation with the fresh sEMG stream of a wristband, serves every vision delay up to 300 ms with a single model, and re-anchors each forecast to the newest pose so that errors do not accumulate. On unseen emg2pose users and in three contact tasks with deadlines in simulation, MKF recovers most of the tracking error that latency adds and, at the largest delay, turns most latency-induced task failures back into successes; the EMG gain persists under tracking noise measured on real egocentric chains. Our evaluation replays recorded operators in a simulated testbed with injected latency, so sim-to-real transfer is untested; a human-in-the-loop study on a real robot is the direct next step. We hope this work encourages the use of the commodity sEMG wristband as a standard robotics modality, first for teleoperation latency and then for the demonstrations that train manipulation policies.

Reproducibility statement

All experiments use the public emg2pose benchmark with its official splits, forward kinematics and metrics; the vision-degradation statistics of Table 4 are measured on the public HOT3D-Clips and EgoEMG releases with their ground-truth hand labels (Appendix B.3). The model and training setup are described in Sec. 3 and Appendix A, the evaluation protocol of the task suite in Appendix B, and the full result tables in Appendix D. Upon acceptance, we will release all code together with the training and evaluation configurations of every reported run. The official emg2pose code is used in accordance with its license.

Ethics statement

This work uses only public datasets collected with consent (emg2pose, HOT3D and EgoEMG); no new human-subject data was collected. EMG signals are biometric; our models consume short causal windows for motor prediction and do not perform identification. Improved teleoperation lowers the cost of robot demonstration data, with the usual dual-use considerations of dexterous robotics.

AI use statement

Large language models were used to assist in drafting and revising the manuscript text. All research ideas, experimental design, data collection, and results are the authors' own; the authors verified all content and take full responsibility for it.

References

Diar Abdlkarim, Massimiliano Di Luca, Poppy Aves, Mohamed Maaroufi, Sang-Hoon Yeo, R. Chris Miall, Peter Holland, and Joseph M. Galea. A methodological framework to assess the accuracy of virtual reality hand-tracking systems: A case study with the Meta Quest 2. *Behavior Research Methods*, 56:1052–1063, 2024.

Prithviraj Banerjee, Sindi Shkodrani, Pierre Moulon, Shreyas Hampali, Shangchen Han, Fan Zhang, Linguang Zhang, Jade Fountain, Edward Miller, Selen Basol, Richard Newcombe, Robert Wang, Jakob Julian Engel, and Tomas Hodan. HOT3D: Hand and object tracking in 3D from egocentric multi-view videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7061–7071, 2025.

Antal K. Bejczy, Won S. Kim, and Steven C. Venema. The phantom robot: predictive displays for teleoperation with time delay. In *IEEE International Conference on Robotics and Automation (ICRA)*, 1990.

Alessandra Bernardini, Roberto Meattini, Alex Pasquali, Gianluca Laudante, Cosimo Gentile, Emanuele Gruppioni, Gianluca Palli, and Claudio Melchiorri. Hidden Markov model-based shared autonomy for grip strength regulation in sEMG driven robot hand control. *IEEE Transactions on Robotics*, 42:2249–2268, 2026.

Kevin Black, Manuel Y. Galliker, and Sergey Levine. Real-time execution of action chunking flow policies. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Jan Bohnerth, Janis Sprenger, Selvakumar Panneer, Björn Browatzki, Anindita Ghosh, and Philipp Slusallek. Evaluating pose forecasting for compensating network latency in full body movements. In *Virtual Reality and Mixed Reality (EuroXR 2025)*, volume 16101 of *Lecture Notes in Computer Science*, pp. 40–61. Springer, 2025.

Xu Chi, Chao Zhang, Yang Su, Lingfeng Dou, Fujia Yang, Jiakuo Zhao, Haoyu Zhou, Xiaoyou Jia, Yong Zhou, and Shan An. Open TeleDex: A hardware-agnostic teleoperation system for imitation learning based dexterous manipulation. *arXiv preprint arXiv:2510.14771*, 2025.

Wenhui Cui, Christopher M. Sandino, Hadi Pouransari, Ran Liu, Juri Minxha, Ellen L. Zippi, Erdrin Azemi, and Behrooz Mahasseni. EMBridge: Enhancing gesture generalization from EMG signals through cross-modal representation learning. In *International Conference on Learning Representations (ICLR)*, 2026.

Alessandro Del Vecchio, Andres Úbeda, Massimo Sartori, José M. Azorín, Francesco Felici, and Dario Farina. Central nervous system modulates the neuromechanical delay in a broad range for the control of muscle force. *Journal of Applied Physiology*, 125(5):1404–1410, 2018.

Joseph DelPreto and Daniela Rus. Sharing the load: Human-robot team lifting using muscle activity. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2019.

Davide Falanga, Suseong Kim, and Davide Scaramuzza. How fast is too fast? The role of perception latency in high-speed sense and avoid. *IEEE Robotics and Automation Letters*, 4(2):1884–1891, 2019.

Aberham Genetu Feleke, Luzheng Bi, and Weijie Fei. EMG-based 3D hand motor intention prediction for information transfer from human to robot. *Sensors*, 21(4):1316, 2021.

William R. Ferrell. Remote manipulation with transmission delay. *IEEE Transactions on Human Factors in Electronics*, HFE-6:24–32, 1965.

Sofia Gilardini, Chenfei Ma, and Kianoush Nazarpour. KinEMbed: Decoding kinematics from electromyography via cross-modal contrastive learning. In *ICML 2026 Workshop on Structured Data for Health*, 2026.

Eric Godden, William Steedman, and Matthew K. X. J. Pan. Robotic characterization of markerless hand-tracking on Meta Quest Pro and Quest 3 virtual reality headsets. *IEEE Transactions on Visualization and Computer Graphics*, 31(5):3025–3034, 2025.

Ankur Handa, Karl Van Wyk, Wei Yang, Jacky Liang, Yu-Wei Chao, Qian Wan, Stan Birchfield, Nathan Ratliff, and Dieter Fox. DexPilot: Vision-based teleoperation of dexterous robotic hand-arm system. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2020.

Masashi Hatano, Zhifan Zhu, Hideo Saito, and Dima Damen. The invisible EgoHand: 3D hand forecasting through EgoBody pose estimation. *arXiv preprint arXiv:2504.08654*, 2025.

Botao He, Zhi Wang, Linna Kuang, Ishaan Ghosh, Jitendra Malik, Cornelia Fermuller, Tingfan Wu, Jiayuan Mao, Ruoshi Liu, Haozhi Qi, and Yiannis Aloimonos. ForceBand: Learning forceful manipulation with sEMG. *arXiv preprint arXiv:2606.26093*, 2026.

Ryan Hoque, Peide Huang, David J. Yoon, Mouli Sivapurapu, and Jian Zhang. EgoDex: Learning dexterous manipulation from large-scale egocentric video. In *International Conference on Learning Representations (ICLR)*, 2026.

Patrick Kaifosh, Thomas R. Reardon, and CTRL-labs at Reality Labs. A generic non-invasive neuromotor interface for human-computer interaction. *Nature*, 645:702–711, 2025.

Jin-Woo Kang and Young-Dae Hong. Upper limb modeling for delay-compensated motion prediction using dilated convolution based on sEMG and IMU sensor fusion. *Journal of Intelligent & Robotic Systems*, 111:119, 2025.

Jin-Woo Kang, Su-Young Wang, and Young-Dae Hong. Upper limb motion prediction for delay compensation in robotic arms using separated neural networks with EMG and IMU fusion. *Robotics and Autonomous Systems*, 202:105475, 2026.

Aws Khalil and Jaerock Kwon. Towards generative predictive display for vision-based teleoperation: A zero-shot benchmark of off-the-shelf video models. *arXiv preprint arXiv:2605.09670*, 2026a.

Aws Khalil and Jaerock Kwon. Nonlinear performance degradation of vision-based teleoperation under network latency. *arXiv preprint arXiv:2603.06850*, 2026b.

Alexander Khazatsky, Karl Pertsch, et al. DROID: A large-scale in-the-wild robot manipulation dataset. In *Robotics: Science and Systems (RSS)*, 2024.

Seok Joon Kim, Junho Lee, Federica Spinola, Taein Kwon, and Mohsen Moghaddam. AHEAD: Anticipatory hand-driven teleoperation via human intent prediction. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2026a.

Sunwoo Kim, Jeongjun Kim, and Joseph J. Lim. Time optimal execution of action chunk policies beyond demonstration speed. In *International Conference on Learning Representations (ICLR)*, 2026b.

Steven M. LaValle, Anna Yershova, Max Katsev, and Michael Antonov. Head tracking for the Oculus Rift. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 187–194, 2014.

Jiahang Liu, Zhipeng Li, Renzhen Le, Xiao Zhang, Naohiko Sugita, Zhenzhi Ying, and Liming Shu. Precise hand-arm teleoperation of dexterous robotic manipulator via markerless vision and high density sEMG fusion. *IEEE Robotics and Automation Letters*, 11(2):1938–1945, 2026.

Yurika Machida and Wataru Kameyama. Two-dimensional arm motion prediction using sEMG and LSTM for teleoperation delay compensation. In *IEEE International Conference on Consumer Electronics (ICCE)*, 2026.

Julieta Martinez, Michael J. Black, and Javier Romero. On human motion prediction using recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2891–2900, 2017.

Rolandos Alexandros Potamias, Jinglei Zhang, Jiankang Deng, and Stefanos Zafeiriou. WiLoR: End-to-end 3D hand localization and reconstruction in-the-wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12242–12254, 2025.

Aditya Prakash, David Forsyth, and Saurabh Gupta. Bimanual 3D hand motion and articulation forecasting in everyday images. *arXiv preprint arXiv:2510.06145*, 2025.

Yuzhe Qin, Wei Yang, Binghao Huang, Karl Van Wyk, Hao Su, Xiaolong Wang, Yu-Wei Chao, and Dieter Fox. AnyTeleop: A general vision-based dexterous robot arm-hand teleoperation system. In *Robotics: Science and Systems (RSS)*, 2023.

Sasha Salter, Richard Warren, Collin Schlager, Adrian Spurr, Shangchen Han, Rohin Bhasin, Yujun Cai, Peter Walkington, Anuoluwapo Bolarinwa, Robert Wang, Nathan Danielson, Josh Merel, Eftychios Pnevmatikakis, and Jesse Marshall. emg2pose: A large and diverse benchmark for surface electromyographic hand pose estimation. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2024.

Thomas B. Sheridan and William R. Ferrell. Remote manipulative control with transmission delay. *IEEE Transactions on Human Factors in Electronics*, HFE-4(1):25–29, 1963.

Zhaofeng Shi, Heqian Qiu, Lanxiao Wang, Xiang Li, and Hongliang Li. Exo2EgoPose: Leveraging exocentric demonstrations for vision-language guided egocentric 3D hand pose forecasting. In *ACM International Conference on Multimedia (ACM MM)*, 2026.

Christian Smith and Patric Jensfelt. A predictor for operator input for time-delayed teleoperation. *Mechatronics*, 20(7):778–786, 2010.

Haoxuan Wang, Gengyu Zhang, Yan Yan, Yuzhang Shang, Ramana Rao Kompella, and Gaowen Liu. Real-time robot execution with masked action chunking. In *International Conference on Learning Representations (ICLR)*, 2026.

Dong Wei, Xiaoning Sun, Xizhan Gao, Shengxiang Hu, and Huaijiang Sun. ALIEN: Implicit neural representations for human motion prediction under arbitrary latency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1861–1870, 2025.

Ziheng Xi, Jiayi Yu, Yitao Wang, Yanbo Duan, Jianjiang Feng, and Jie Zhou. EgoEMG: A multimodal egocentric dataset with bilateral EMG and vision for hand pose estimation. *arXiv preprint arXiv:2605.05712*, 2026.

Dezhen Xiong, Xin Fu, Daohui Zhang, Yaqi Chu, Yiwen Zhao, and Xingang Zhao. Robotic telemanipulation with EMG-driven strategy-assisted shared control method. *Science China Technological Sciences*, 67:3812–3824, 2024.

Zhao-Heng Yin, Changhao Wang, Luis Pineda, Krishna Bodduluri, Tingfan Wu, Pieter Abbeel, and Mustafa Mukadam. Geometric retargeting: A principled, ultrafast neural hand retargeting algorithm. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025.

Fan Zhang, Valentin Bazarevsky, Andrey Vakunov, Andrei Tkachenka, George Sung, Chuo-Ling Chang, and Matthias Grundmann. MediaPipe hands: On-device real-time hand tracking. In *CVPR Workshop on Computer Vision for Augmented and Virtual Reality*, 2020.

Qianyou Zhao, Wenqiao Li, Chiyu Wang, and Kaifeng Zhang. DexEMG: Towards dexterous teleoperation system via EMG2Pose generalization. *arXiv preprint arXiv:2603.05861*, 2026.

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Robotics: Science and Systems (RSS)*, 2023.

APPENDIX

A Implementation Details **13**

B Additional Evaluation Details **14**

B.1 Figure construction 14

B.2 Task-suite protocol 15

B.3 Calibration of the vision degradation on real egocentric chains 15

C Release Task Details and Failure Modes **16**

C.1 Timed release and the shape-limited variant 16

C.2 Moving catch: arrival-speed sweep 17

D Additional Results and Full Tables **18**

E Detection Before Motion Onset **19**

F Latency Budget **20**

A IMPLEMENTATION DETAILS

Data. emg2pose (Salter et al., 2024) contains 25,253 HDF5 files with 16-channel 2 kHz sEMG and 20 joint-angle labels; the official metadata split the data along the user, stage (gesture) and user×stage axes. Each window contains a 1 s pose history at 50 Hz and EMG features at 100 Hz, and the prediction target spans 0.5 s at 50 Hz. The window stride is 200 ms for training and 500 ms for evaluation.

Models. Kin-only is an 8-layer causal Transformer (d_{model}=256, 8 heads) with a residual output relative to the last observed pose (Martinez et al., 2017) and 6.72M parameters. MKF has a 6-layer trunk, a 2-layer EMG encoder (self-attention within the 1 s window, which ends at t, so no future data is visible) and interleaved cross-attention, with 8.41M parameters. Both models take the normalized latency $L/300$ as an input token, and $L \sim \mathcal{U}\{0, 50, \ldots, 300\}$ ms is sampled for each training example. We train with AdamW (learning rate 3×10^{-4}, batch size 256, mixed precision) for 50k steps, with two seeds for the headline numbers. The D-CNN baseline follows the dilated-convolution architecture of Kang & Hong (2025); our implementation uses dilation rates 1/2/4/8, whereas the original uses 1/1/2/2/2/3. It receives the same inputs as MKF (EMG tokens, upsampled pose history and latency token) and is evaluated at the original size of 0.05M parameters and at a width-scaled size of 7.7M.

EMG features. The EMG is band-pass filtered at 20–450 Hz, notch filtered at 50 Hz, converted to a causal 20 ms moving RMS at 100 Hz, and normalized per window. Like any transport delay of the wristband, the causal RMS window shifts the differential latency slightly below the nominal L. Encoder ablations on the validation split at L=300 ms give 9.57° for the envelope, 9.71° for a 6-band log-spectral representation, 9.70° with electrode-rotation augmentation and 9.83° for their combination; a learned front-end on the raw 2 kHz EMG gives similar values.

Temporal grid. Joint angles in emg2pose are stored on the 2 kHz EMG time base. For a window at time t and latency L, the pose history is cut at sample $t - L$ and subsampled every 40 samples, so its 50 frames lie exactly at $t - L - 20k$ ms (k=0, . . . , 49) and $\mathbf{q}(t - L)$ is the sample at $t - L$. The EMG window is $[t-1\,\mathrm{s}, t]$ on the same time base, and the targets lie at $t+20k$ ms. Since the outputs lie on a 20 ms grid, the shortest horizon in Fig. 3b is 40 ms, and every reported horizon and latency is a multiple of 20 ms. Both streams carry the same sinusoidal positional encoding over their 50 tokens (Fig. 2). The offset L between the two time axes is not encoded in the positions; it is supplied to every pose token as $L/300$, and the cross-attention learns the latency-dependent alignment. Two observations indicate that this alignment generalizes between the seven training latencies: the gain grows monotonically with L (Table 1), and a misspecified L has little effect (Table 3; an error of 25 ms changes the unseen-user MAE at L=100 ms by less than 1%).

Normalization and loss. Angles are standardized with the global per-joint mean and standard deviation of the training split, and ρ is the smooth-ℓ_1 penalty with transition point β=1 in standardized units. The velocity feature of the earliest history frame is set to zero.

EMG-only forecaster. The EMG-only forecaster of Table 2 consists of the EMG path of MKF (front-end, stride-2 stem and two self-attention blocks) followed by the same six-block trunk without cross-attention. It reads the last token and predicts the 25 future poses as absolute angles, since no stale pose is available as an anchor; it has 6.8M parameters and is trained with the recipe of MKF. Its task differs from that of the emg2pose benchmark, which estimates the current pose: it forecasts the pose 100 ms ahead from one second of 100 Hz envelopes. Its error of 13.5° at Δ=100 ms lies within the range of 13.3–14.0° reached by the released regression decoders under the same scoring (Table 2).

Motion states and data scaling. At L=300 ms, the ordering no compensation > Kin-only > MKF holds for every motion state: 8.09, 7.28 and 5.74° at onsets, 12.84, 10.06 and 8.55° during steady motion, and 11.08, 9.22 and 8.00° at reversals. At L=0 during rest, holding the pose (0.55°) is more accurate than both forecasters (Kin-only 0.89°, MKF 0.78°). The EMG gain saturates at about 40 training users (about 25% of the training set); additional users improve Kin-only and MKF equally without widening the gap (Table 8).

Per-user fine-tuning. Fine-tuning MKF, either fully or with a frozen trunk, on the first sessions of a held-out user did not improve on the cross-user model on the later sessions of that user. This is consistent with the saturation of the gain at about 40 training users and with the small difference between within-subject and cross-user results (Table 13).

B Additional Evaluation Details

B.1 Figure construction

Fig. 1: the recording is a randomly selected held-out-user test file of the L=300 ms export. Within a 4 s fast-motion window, the figure shows the three ticks, at least 0.3 s apart, with the largest difference between the error without compensation and the error of MKF. This selection favors MKF by construction; over the whole sequence (valid frames), the error is 19.7° without compensation and 10.9° with MKF. The operator and the wristband in panel (a) are illustrations. Fig. 5: the scenes are rendered with a static UR5e arm posed by inverse kinematics behind the fixed wrist, a partner gripper, a guard-door frame with a stack light and a transparent bin. All added geometry is either visual only or a kinematic body that never touches the hand or the object (the lift gate is the collision cover of the task, drawn translucently), and every rendered trial was checked to reproduce the recorded outcome. Fig. 8: the four exported sequences are the test files with the highest motion energy; their effective lags, verified by cross-correlation, are 220 and 320 ms including the 20 ms tick. Commands are mapped to the Shadow Hand joint by joint (PIP and DIP individually); the four

coupled J0 actuators are tendon transmissions, and each rollout starts by holding the first command under dynamics for 1 s.

B.2 TASK-SUITE PROTOCOL

Common protocol. Closure onsets are detected on the mean MCP flexion of the four fingers: the hand is open below 25% of the range for at least 0.3 s, then crosses this threshold and reaches 65% within 0.5 s. Opening onsets are defined symmetrically (closed above 75% for at least 0.7 s, then below 35%), and events are at least 3 s apart. Left- and right-hand recordings use the corresponding Shadow Hand model. The benchmark replays 51 randomly selected held-out-user test recordings (fixed seed; identical file sets for Kin-only and MKF). The export of each latency carries its own timeline aligned to the ground truth, and the event tick is mapped into each export by exact array alignment. Every trial starts 1.2 s (release: 0.7 s) before the event with a 1 s dynamic warm start. As in the continuous hand tracking of Sec. 5.1, every condition executes the 20 ms slice of its forecast at each tick. emg2pose stores frames in which the motion-capture IK solver failed as all-zero joint vectors, and the official protocol skips them; any event whose trial window contains such a frame is excluded before scoring (89 of 300 detected closures). Since emg2pose contains free-space gestures without objects, each task instruments the recorded motion with a virtual object that is timed (and, for the release, placed) by the operator's own recording; the pure-delay calibration curves of Fig. 5 show how each task responds to a delay of the operator's own motion. *Catch.* The ball has a radius of 2.1 cm, a mass of 50 g and a bouncy contact (MuJoCo `solref` damping ratio 0.25). It is released from rest 8 cm above the grasp point 60 ms (three ticks) before the onset, so that it lands about 68 ms after the onset after a free fall of 128 ms. A trial succeeds if, 0.8 s after the onset, the ball is within 12 cm of the grasp point and above the floor. The four free parameters are varied in a factorial design on a pilot subset of 4 recordings (Table 14; $n{\approx}21$ per setting; 27 simulated grids $\times$ 3 post-hoc thresholds); MKF outperforms both baselines in 81/81 settings. *Gate.* A kinematic plate (half-width 11 cm, half-height 6.5 cm) moves at 0.3 m/s along the finger axis from 62 cm and stops at 41.5 cm from the wrist, 1–3.5 cm beyond the closed fist (extended fingertips reach 45 cm). Its arrival is timed 200 ms after the first tick at which the operator's own replayed hand is entirely inside the stop line, and a trial fails if any contact force between hand and plate exceeds 1 N while the plate is moving. The 200 ms margin matches the pure-delay tolerance of the catch task, and at $L{=}300$ ms the ordering in the gate task holds at every setting of Table 15. *Release tasks.* See Appendix C.

B.3 CALIBRATION OF THE VISION DEGRADATION ON REAL EGOCENTRIC CHAINS

HOT3D chains. We use 64 clips of the Aria training split of HOT3D-Clips (Banerjee et al., 2025) (every 23rd clip identifier; 150 frames at 30 Hz each), whose hands are annotated with a motion-capture system in the UmeTrack parameterization, which has the same 20 joint angles as the emg2pose labels. The fisheye RGB stream of the head-mounted Aria glasses is rectified to a 90° pinhole view of 1024 pixels. Each tracker is run on every frame, MediaPipe Hands (Zhang et al., 2020) in video mode and WiLoR (Potamias et al., 2025) with its released detector and weights. A detection is assigned to a ground-truth hand when its wrist projects within 12% of the image width of that hand's wrist, and a hand counts as present when the ground truth marks it visible in the RGB stream. Both trackers output 21 landmarks, so the tracked and the ground-truth landmarks are converted to joint angles by the same inverse-kinematics fit of the UmeTrack hand model (wrist pose, scale and joint angles; gradient descent followed by quasi-Newton refinement), and the error of a frame is the difference between the two fits. The error of the fit itself is therefore not attributed to the tracker; on ground-truth landmarks the fit has a median angle error of 5.6° and a landmark residual of 1.7 mm, compared with residuals of 7.0 mm (MediaPipe) and 4.1 mm (WiLoR) on tracked landmarks. For each hand-clip we compute the mean absolute error, the per-joint standard deviation σ averaged over joints, the lag-one autocorrelation of the error (converted to the AR(1) time constant τ), the frame-to-frame standard deviation of the error (jitter), and the runs of consecutive frames without a detection. Table 4 reports medians over the 103 (MediaPipe) and 107 (WiLoR) valid hand-clips and the pooled loss statistics. The error of both trackers depends little on hand speed ($25.0/27.4/27.2/27.1^\circ$ for MediaPipe and $15.4/14.8/14.1/13.6^\circ$ for WiLoR across speed quartiles). For scale, a landmark error of about one centimetre on a phalanx about three centimetres long corresponds to an angle error of roughly 20°.

EgoEMG chain. The released vision-only pose estimator of EgoEMG (Xi et al., 2026), a ResNet-18 applied to head-camera hand crops, is run on the 12 subjects that the released split files mark as test subjects and compared with the motion-capture labels of the dataset on a 100 Hz grid. The released crops are located by projecting the motion-capture keypoints into the head camera (Xi et al., 2026), so this chain contains no hand detection; frames without a crop, in which these keypoints are missing, count as tracking losses. Because this estimator is trained on data from the same camera and recording setup, its error is smaller and far more correlated in time than that of the general-purpose trackers (Table 4), and its tracking losses are rarer but longer (mean 2920 ms).

Retraining. For each chain, the pose stream of emg2pose is degraded with the model used for Table 9: a zero-order hold at the frame rate of the chain, AR(1) noise with the measured σ and τ (rounded to 27° and 85 ms for MediaPipe, 9° and 150 ms for WiLoR, and 7.9° and 515 ms for EgoEMG), and one freeze per one-second window with probability 0.15, 0.2 and 0.005, respectively, lasting 100–1000 ms (the longer EgoEMG losses are capped so that each window retains some fresh pose). Kin-only and MKF are trained with the recipe and budget of Table 9 (one seed) and evaluated with the same exclusion of inverse-kinematics failures. The uncompensated row holds the newest degraded pose and therefore contains the tracker noise. The synthetic degradation of Table 9 (5°, 100 ms, 30 Hz, 100–300 ms freezes in 30% of windows) was set before these measurements.

C Release Task Details and Failure Modes

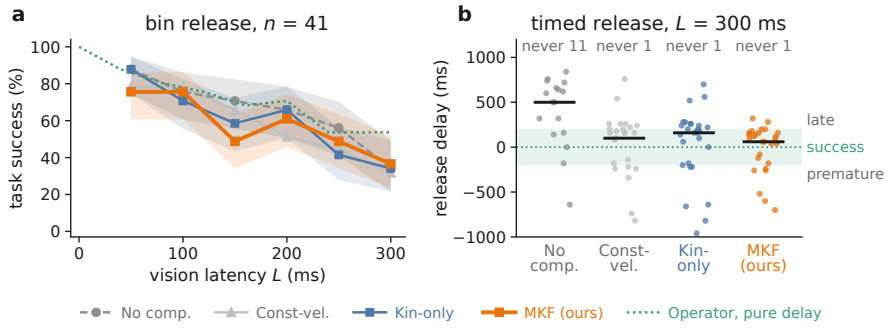

Figure 6: **Release tasks. (a)** Shape-limited bin version: the ball held on a palm tilted by 45° rolls off the fingertips at the opening onset into a 28×10 cm bin that slides past at 0.15 m/s and is centred, from the operator's own replay, under the operator's landing point at the operator's landing time. **(b)** Timed release at L=300 ms: time at which the ball of each condition crosses the hatch plane relative to the operator (green band: success window; bars: medians; balls never released are counted above the axis).

Table 5: **Release tasks** (test, unseen users). Success rates, and for the timing-only hatch version the failure-mode breakdown at L=300 ms: late (>200 ms after the operator), premature (>200 ms before), never released within the trial, and the median release delay relative to the operator.

	Bin (v1), n=41			Hatch (v2), n=30			Hatch failure modes, L=300			
Condition	100	200	300	100	200	300	late	premature	never	median (ms)
No compensation	76%	66%	34%	90%	23%	13%	12	3	11	+500
Const.-velocity	78%	51%	32%	90%	73%	37%	7	11	1	+100
Kin-only	71%	66%	34%	87%	67%	40%	10	7	1	+160
MKF (ours)	76%	61%	37%	90%	80%	67%	2	7	1	+60

C.1 Timed Release and the Shape-Limited Variant

Setup. The palm-up hand is pitched by 45° with the fingertips down. A ball with a radius of 2.5 cm placed on the palm rests against the curled fingers and rolls off when the hand opens. Events are

opening onsets preceded by at least 1.1 s of closed hand, so that every command stream, including one that is 320 ms old, still holds the ball when the trial starts; there are n=30 (hatch) and 41 (bin) operator-feasible events. *Timed release (hatch; main text).* Only the time at which the ball crosses the hatch plane matters. The 30×30 cm hatch is open within ±200 ms of the operator's own crossing, with 100 ms ramps at both edges, so a premature drop lands on the closed cover in the same way as a late one. By construction, its pure-delay calibration is a step at the edge of the window (Fig. 5, bottom, dotted). At L=300 ms (Table 5) the uncompensated release is +500 ms late at the median, and 11 of 30 balls are never released within the trial; MKF releases at a median delay of +60 ms and is too late in only 2 events (Kin-only: 10; no compensation: 12). The success of MKF is limited to 67% by premature drops: in 7 (MKF), 7 (Kin-only) and 11 (constant velocity) events, the predicted closed pose is slightly more open than the operator's, and the ball rolls out while the operator is still holding it (Fig. 6b). This failure corresponds to the cost of forecasting during rest reported in Appendix A: during a static hold, the forecaster predicts small motions that the operator does not make. Gating the forecast by motion state could reduce this cost, which we leave to future work. *Bin release (shape-limited).* The first version of the task timed and placed a sliding bin from the operator's own replay. Its outcome depends on the shape of the release more than on its timing: delaying the operator's own motion by only 40 ms already costs 12% of the successes, because the finger that lowers first sets the direction in which the ball rolls, and a 10 cm bin cannot distinguish a lateral drift of 3 cm from a delay of 200 ms. No forecaster improves on no compensation at any latency or at any of nine bin speeds and widths (Fig. 6a; at L=300 ms: no compensation 34%, Kin-only 34%, MKF 37%; Table 15). We include this task as an example in which timing information alone does not determine success.

C.2 Moving catch: arrival-speed sweep

Setup. To tighten the deadline of the catch task without changing its rules, the ball is delivered with a horizontal speed v across the palm. It lands at the same point and time as in the main task and, unless it is secured, rolls off the fingertips; v=0 reproduces the main task. Feasibility decreases with speed even for the operator (n=152 to 70 events from v=0 to 1 m/s), and the time window narrows: the operator's own motion delayed by 100 ms still succeeds in 93% of events at v=0 and in 37% at v=1 m/s. As shown in Fig. 7, at L=300 ms the success rate without compensation falls from 57% to 14%, while MKF maintains 92% to 54% and stays above Kin-only (39% at v=1 m/s) at every speed. At L=200 ms, Kin-only is ahead of MKF at one grid point (v=0.75 m/s, 56% vs. 52%, within the Wilson bands).

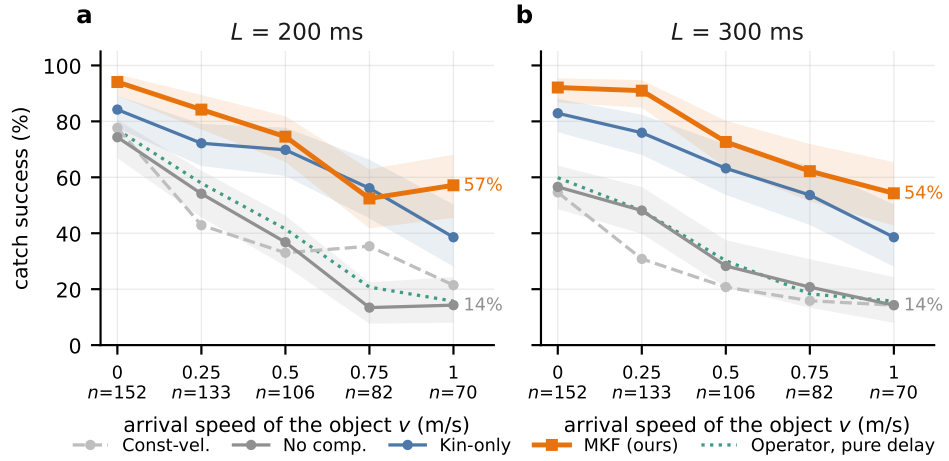

Figure 7: **Moving catch: success versus arrival speed.** Catch success versus the horizontal speed at which the ball reaches the palm, at L=200 and 300 ms (Wilson 95% bands; n: operator-feasible events per speed). The dotted curve is the operator's own motion delayed by L.

Table 6: **Continuous hand tracking over a 900-file sample of the test split** (stratified over the three axes, re-prediction every 20 ms tick): per-tick MAE (deg), measured phase lag (ms), and error slope (deg/min). Velocity integration: MKF increments accumulated without per-tick re-anchoring.

	$L = 100$ ms			$L = 200$ ms			$L = 300$ ms		
Condition	MAE	lag	slope	MAE	lag	slope	MAE	lag	slope
No compensation	4.38	120	0.33	6.44	220	0.54	7.66	320	0.53
Const.-velocity	6.34	93	0.64	12.27	190	1.17	18.03	287	1.78
Kin-only	3.48	106	0.20	5.00	205	0.37	5.90	310	0.44
MKF (ours)	**3.24**	**104**	**0.19**	**4.40**	**204**	**0.17**	**5.10**	**307**	**0.15**
MKF, velocity integration	112.55	0	170	123.86	0	179	123.56	0	188

D Additional Results and Full Tables

Table 7: **Synchronized-delay control** (test, unseen users, Δ=100 ms; MAE in deg / gain over Kin-only). Top: the fresh-EMG MKF of Table 1 (seed 0) at L=200 ms with its EMG stream delayed by e_L at test; gains are relative to the Kin-only model of the same seed (6.07°). Bottom: a model trained with e_L=L, tested with the same synchronized delay and with fresh EMG; gains are relative to the two-seed Kin-only of Table 1.

Trained on fresh EMG; L=200 ms, EMG delayed by e_L at test			
e_L	0 ms (fresh)	100 ms	200 ms (= L)
MKF error / gain	5.12 / +15.6%	5.65 / +6.8%	6.34 / −4.5%
Trained with e_L=L (EMG as stale as the pose)			
	L=100 ms	L=200 ms	L=300 ms
Kin-only (two seeds)	4.94	6.05	6.69
MKF, fresh EMG (Table 1)	4.35 / +12.1%	5.14 / +15.1%	5.66 / +15.4%
MKF, trained and tested with e_L=L	4.71 / +4.7%	5.84 / +3.4%	6.53 / +2.3%
same model, tested with fresh EMG	4.81 / +2.6%	6.03 / +0.3%	6.72 / −0.5%

Table 8: **EMG gain versus number of training users** (validation split, Δ=100 ms; seed 0, recorded under the earlier evaluation without IK-failure masking, so absolute values are above those of Table 1). Cells: Kin-only / MKF joint-angle MAE (deg) and the EMG gain.

Training users	L=0	L=100 ms	L=200 ms	L=300 ms
∼40 (25%)	5.39 / 5.16 (+4.3%)	8.85 / 7.78 (+12.1%)	11.00 / 9.23 (+16.1%)	12.16 / 10.07 (+17.2%)
79 (50%)	5.21 / 4.96 (+4.8%)	8.64 / 7.47 (+13.5%)	10.74 / 8.90 (+17.1%)	11.80 / 9.78 (+17.1%)
158 (100%)	5.16 / 4.89 (+5.2%)	8.52 / 7.39 (+13.3%)	10.55 / 8.79 (+16.7%)	11.63 / 9.57 (+17.7%)

Official decoders as command streams (Table 2). The five released emg2pose checkpoints are loaded with the official code and evaluated, under their own protocols, on the 6,165 test files that fit their window of 11,790 samples. The tracking models receive the true pose at the start of every window and the regression models receive none. Predictions are resampled to the label rate, samples without a valid label at t or at $t+\Delta$ are excluded, and the current-pose output at t is scored against the operator's pose at $t+\Delta$. At Δ=0, this scoring gives 10.5° pooled over the three test axes for the released tracking vemg2pose model, compared with 10.0° from the official evaluation script on the same checkpoint; per-user and per-sample averaging change the user-axis value by less than 0.1°. The EMGFormer entries are the user-split values reported by Xi et al. (2026) and were not re-run. *Time-shuffle control.* The shuffled model and the Kin-only and MKF values reported with it in Sec. 4 were trained with an earlier recipe and evaluated together with inverse-kinematics failures excluded, so they are comparable with each other and not with the two-seed models of Table 1. *Synthetic degradation and tracking loss.* With inverse-kinematics failures excluded, the gains with the synthetic degradation of Table 9 are +4.8/+9.5/+11.8/+12.1% at L=0/100/200/300 ms. In a tracking-loss stress test with the same models (freezes extended to the newest frame, L=200 ms, tick by tick), the error within frozen windows is 11.70° without compensation, 9.47° for Kin-only and 8.54° for MKF. *Servo dynamics (Fig. 8).* Relative to no compensation, MKF reduces the executed

Table 9: **Accuracy under synthetic vision degradation** (test, unseen users, Δ=100 ms). Kin-only and MKF are re-trained on a pose stream degraded by a 30 Hz zero-order hold, AR(1) noise of 5° magnitude and 100 ms time constant, and random 100–300 ms tracking-loss freezes in 30% of windows. Clean-input columns: the two-seed models of Table 1.

	clean (mocap) input				vision-quality input			
Method	L=0	100	200	300	L=0	100	200	300
No compensation	3.91	6.31	7.72	8.41	6.23	8.21	9.42	10.01
Kin-only	3.10	4.94	6.05	6.69	4.64	6.06	6.88	7.36
MKF (ours)	**2.93**	**4.35**	**5.14**	**5.66**	**4.42**	**5.48**	**6.07**	**6.47**
EMG gain (MKF vs. Kin-only)	+5.6%	+12.1%	+15.1%	+15.4%	+4.8%	+9.5%	+11.8%	+12.1%

Table 10: **Real-time budget** (RTX 3090, batch 1, CUDA events, 1000 iters). The effective lead is $\text{lead}_{\text{eff}} = \text{EMD} + \Delta - t_{\text{inf}}$.

Model	params	mean (ms)	P50 (ms)	P99 (ms)
Kin-only (kinematics-only)	6.72M	6.4	5.8	13.7
MKF (kinematics+EMG)	8.41M	8.5	8.3	16.4
EMG featurization (full 1 s window)	–	1.04	0.93	2.08

Table 11: **The task suite.** Design constants of the three contact tasks of Sec. 5.2; the shape-limited bin release is described in Appendix C.

Task	Object / hand	Anchor event	Deadline (why late fails)	Success	n / events
Catch	elastic ball (r 2.1 cm, 50 g) dropped 8 cm onto the palm-up hand	closure onset; ball released 60 ms before it, arrives at onset+68 ms	open palm sheds the bouncy ball within a few hundred ms	ball within 12 cm of the grasp point and off the floor at onset+0.8 s	152 / 211
Gate	plate closing at 0.3 m/s towards the fingertips of the palm-up hand	closure onset; plate reaches its stop 200 ms after the operator's hand has cleared the line	fingers still extended are struck (contact force >1 N while the plate moves)	no contact	47 / 211
Release	2.5 cm ball resting against the curled fingers of a 45°-tilted palm; 30×30 cm hatch 25 cm below	opening onset after ≥1.1 s of closed hold; hatch open only within ±200 ms of the operator's own crossing	a late (or premature) release lands on the closed cover	ball crosses the hatch plane within the window	30 / 48

fingertip error by 32% at L=200 ms and by 36% at L=300 ms. The data-scaling sweep (Table 8) and Fig. 8 were recorded with the earlier evaluation without IK-failure masking; their absolute values include these frames, and the orderings are unaffected. *Phase on the recording of Fig. 1.* For the fastest joint of this recording, the uncompensated command is the operator's motion delayed by 300 ms; it has the same shape and is anti-correlated with the operator in time ($r = -0.71$, mean error 53°). The forecast of MKF has a smaller amplitude (peak-to-peak 100° vs. 118°) and is in phase with the operator ($r = +0.80$, mean error 16°).

E Detection Before Motion Onset

We tested whether the EMG stream of MKF supports detection before motion onset, that is, predicting that the hand is about to move before it moves. With debounced low-threshold onset labels and a causal detector window no longer than L, at matched false-alarm rates (≤10/min), the detection rate of the EMG model before motion onset is statistically indistinguishable from that of the kinematic

Table 12: **Accuracy on the three generalization axes** (test, Δ=100 ms). Joint-angle MAE (deg); Kin-only and MKF are two-seed means, and gains are mean±half-range over seeds.

Axis	Method	$L = 0$	$L = 100$	$L = 200$	$L = 300$
unseen user	no comp.	3.91	6.31	7.72	8.41
	Kin-only	3.10	4.94	6.05	6.69
	MKF	2.93	4.35	5.14	5.66
	gain	$+5.6 \pm 0.1\%$	$+12.1 \pm 0.3\%$	$+15.1 \pm 0.5\%$	$+15.4 \pm 0.4\%$
unseen gesture	no comp.	5.35	8.94	11.20	12.63
	Kin-only	4.09	6.98	8.95	10.22
	MKF	3.84	6.08	7.53	8.53
	gain	$+6.2 \pm 0.4\%$	$+12.9 \pm 0.1\%$	$+15.9 \pm 0.1\%$	$+16.6 \pm 0.2\%$
unseen both	no comp.	5.07	8.25	9.91	10.68
	Kin-only	4.01	6.58	8.11	8.92
	MKF	3.74	5.79	7.03	7.80
	gain	$+6.6 \pm 0.3\%$	$+12.0 \pm 0.1\%$	$+13.3 \pm 0.1\%$	$+12.5 \pm 0.2\%$

Table 13: **Within-subject evaluation** (held-out sessions of *seen* training users; Δ=100 ms).

	$L = 0$	$L = 100$	$L = 200$	$L = 300$
no comp.	3.82	6.26	7.73	8.51
Kin-only	3.11	4.87	5.93	6.53
MKF	2.88	4.25	5.02	5.52
gain (within)	+7.2%	+12.8%	+15.3%	+15.5%
gain (cross-user)	+5.6%	+12.1%	+15.1%	+15.4%

Table 14: **Catch-task sensitivity analysis.** Full factorial over the four free design parameters ($3^4 = 81$ settings; contact elasticity, ball radius with mass at constant density, ball-arrival offset relative to closure onset, and success radius), run on the 4-recording pilot subset with the earlier model export without IK-failure masking. Feasibility is re-established from the operator's own recording in every setting, so n varies. Each row marginalizes over the other three parameters. The reported configuration is starred. Margins are in percentage points (pp).

Parameter	Value	No comp.	MKF (ours)	margin (pp)
contact elasticity (`solref` ζ)	0.20	51%	91%	+41
	0.25*	49%	90%	+41
	0.35	49%	90%	+41
ball radius (mm)	18	49%	94%	+45
	21*	54%	93%	+39
	25	46%	85%	+38
arrival offset (ms)	0	36%	86%	+50
	60*	48%	96%	+48
	120	65%	90%	+25
success radius (cm)	8	40%	84%	+44
	12*	52%	93%	+41
	16	56%	94%	+37

baseline. The anticipatory information measured in the main text therefore acts after motion onset, where it reduces the forecasting error (Sec. 4).

F Latency Budget

Table 16 collects latencies reported in published work, grouped into the two terms of Sec. 3: the vision chain ahead of the interface (L) and everything downstream of it (Δ).

Table 15: **Gate and release sensitivity** (full 51-file set, L=200/300 ms; feasibility re-derived per setting). Reported configurations are starred.

Task	setting	n	$L = 200$: no comp. / Kin-only / MKF	$L = 300$: no comp. / Kin-only / MKF
gate	margin 160 ms, gate 0.3 m/s	52	38 / 75 / 88%	17 / 50 / 77%
	margin 160 ms, gate 0.5 m/s	52	50 / 90 / 92%	27 / 62 / 81%
	margin 200 ms, gate 0.3 m/s*	47	43 / 89 / 91%	26 / 60 / 83%
	margin 200 ms, gate 0.5 m/s	47	66 / 96 / 96%	32 / 66 / 85%
	margin 240 ms, gate 0.3 m/s	43	67 / 98 / 98%	33 / 70 / 86%
	margin 240 ms, gate 0.5 m/s	45	98 / 96 / 96%	36 / 76 / 84%
release	bin 0.10 m/s, width 8 cm	41	63 / 61 / 56%	49 / 29 / 34%
	bin 0.10 m/s, width 10 cm	41	63 / 71 / 56%	54 / 39 / 37%
	bin 0.10 m/s, width 14 cm	41	68 / 76 / 68%	56 / 51 / 54%
	bin 0.15 m/s, width 8 cm	41	56 / 51 / 56%	20 / 29 / 29%
	bin 0.15 m/s, width 10 cm*	41	66 / 66 / 61%	34 / 34 / 37%
	bin 0.15 m/s, width 14 cm	41	73 / 76 / 73%	56 / 51 / 56%
	bin 0.20 m/s, width 8 cm	41	51 / 49 / 54%	15 / 20 / 24%
	bin 0.20 m/s, width 10 cm	41	68 / 54 / 63%	15 / 34 / 29%
	bin 0.20 m/s, width 14 cm	41	73 / 71 / 73%	51 / 44 / 59%

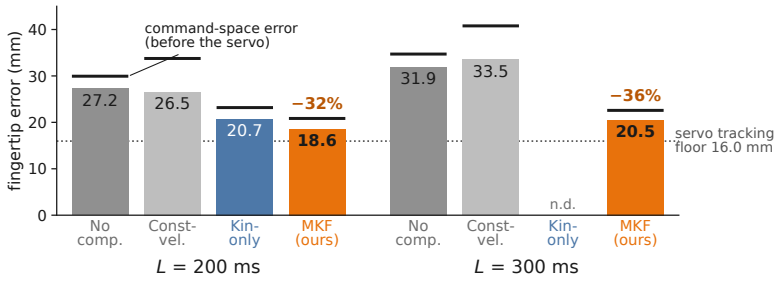

Figure 8: **Execution through servo dynamics** (continuous hand tracking). Four fast-motion test sequences streamed through a MuJoCo Shadow Hand under 500 Hz position servos. The fingertip error of each command stream is measured against a rollout driven by the true pose through the same servos. Black dashes: command-space error before the servo; percentages: change in error of MKF relative to no compensation. n.d.: Kin-only was not exported at L=300 ms in this rollout set.

The first three rows are end-to-end measurements of hand tracking, in which a hand moves along a known trajectory, driven by a manipulator or through a fixed grid of targets, and the tracker output is compared with it. The following rows are single components, and the last two rows are figures for complete systems. The widely used vision-based teleoperation systems do not report an end-to-end latency of the vision chain; they report the rate of the hand stream (25–30 Hz) (Qin et al., 2023; Handa et al., 2020), per-module timings (Qin et al., 2023) or a coarse whole-system figure (Handa et al., 2020).

Four entries require a note. The hand-tracking measurements were made on VR headsets; we use them because the perception problem, markerless hand tracking from cameras on commodity hardware, is the same as in a vision-based teleoperation system. The synchronization figure is the error between sensor streams; a system that waits for time-aligned streams incurs it as waiting time for the freshest stream, so it is an upper bound on this contribution. The video-transport figure was measured on a vehicle-teleoperation testbed over WiFi and serves as an order of magnitude for streaming video over a local wireless link. The camera figures are bounds on the sensing latency of frame-based cameras, derived for obstacle avoidance on a quadrotor (Falanga et al., 2019).

Summing these components, the vision term ranges from tens of milliseconds, for a tracker on a local wired link, to about 300 ms, for the mean tracking latency of 229 ms at motion onset (Quest Pro) plus video transport (62 ms median, 86 ms at the 95th percentile) and synchronization error (43–65 ms), which hand-tracking benchmarks do not include. This sum of separately measured parts defines the range over which we sweep L. Tracking latency also depends strongly on the phase of the motion. Godden et al. (2025) measure 229.2 ms (Quest Pro) and 220.5 ms (Quest 3) at motion onset, when the motion prediction of the tracker has not yet taken effect, and 15.8 and 14.4 ms at constant velocity, after a warm-up of about 660 ms. The delay of the vision chain is therefore largest

at motion onsets; on these segments, MKF reduces the error from 8.09° without compensation to 5.74° at L=300 ms (Sec. 4). On the robot side, the reported values are close to 100 ms, for example a control response of the dexterous hand below 120 ms (Liu et al., 2026) and the 100 ms motor delay compensated by Kang & Hong (2025); we therefore fix Δ=100 ms in the main tables.

Table 16: **Reported latencies relevant to vision-based teleoperation, by stage.** All values are as stated in the cited papers.

Term	Stage	Reported value	Source
L	hand tracking, motion to reported pose	45.0 ms average (Quest 2)	Abdlkarim et al. (2024)
L	hand tracking at motion onset, Quest Pro / Quest 3	229.2 ± 146.3 / 220.5 ± 101.4 ms	Godden et al. (2025)
L	hand tracking at constant velocity, Quest Pro / Quest 3	15.84 ± 9.69 / 14.44 ± 9.50 ms	Godden et al. (2025)
L	camera sensing latency, bounds (monocular / stereo)	26–40 / 17–70 ms	Falanga et al. (2019)
L	hand-keypoint recognition and preprocessing	$<$ 25 ms	Liu et al. (2026)
L	hand-pose inference (RGB / RGB-D)	26 ± 5 / 27 ± 5 ms (desktop) 34 ± 5 / 35 ± 5 ms (laptop)	Qin et al. (2023)
L	pose-stream cadence	25 Hz designed maximum 20 Hz overall system response	Qin et al. (2023) Liu et al. (2026)
L	sensor-stream synchronization error	42.75 ms and 60–65 ms (two setups)	Chi et al. (2025)
L	transport, local wireless	$<$ 2 ms	Liu et al. (2026)
L	transport, streamed video over WiFi	62 ms median, 86 ms p95	Khalil & Kwon (2026b)
Δ	retargeting	9 ± 7 ms (desktop), 10 ± 9 ms (laptop)	Qin et al. (2023)
Δ	arm control response	$<$ 10 ms	Liu et al. (2026)
Δ	dexterous-hand control response	$<$ 120 ms	Liu et al. (2026)
Δ	motor delay compensated in a delay study	100 ms	Kang & Hong (2025)
$L+\Delta$	complete system, local RGB-D and HD-sEMG	$<$ 200 ms total	Liu et al. (2026)
$L+\Delta$	complete system, daisy-chained modules	about one second	Handa et al. (2020)