

RC-MH-GRU untuk Prediksi Arah Pandang 360° pada Telepresensi Robot

Abdullah Iskandar^{1,2}, Pima Hani Safitri^{1,3}, Jati Hiliamsyah Husen⁴, Muhammad Arifin⁵

¹Program Studi Informatika, Universitas Telkom, Kampus Surabaya, Indonesia,
abdullahiskandar@telkomuniversity.ac.id; phanisafitri@telkomuniversity.ac.id

²Center of Excellence on Human Centric Engineering, Universitas Telkom, Indonesia,
abdullahiskandar@telkomuniversity.ac.id

³Center of Excellence on Motion Technology for Safety, Health, and Wellness, Universitas Telkom, Indonesia,
phanisafitri@telkomuniversity.ac.id

⁴Program Studi S1 Rekayasa Perangkat Lunak, Universitas Telkom, Bandung, Indonesia,
jatihusen@telkomuniversity.ac.id

⁵Intelligent and Mechanical Interaction Systems, Graduate School of Science and Technology, University of Tsukuba, Tsukuba, Japan, *muhammad.arifin@ftl.iit.tsukuba.ac.jp*

Keywords:

360-degree video,
Head-direction prediction,
Multi-horizon prediction,
Recurrent neural network,
Viewport prediction,

ABSTRACT

Viewport prediction can help a 360-degree video system prioritize panoramic regions for tile-based, region-of-interest (ROI), or adaptive-quality streaming. This study evaluates a Ridge-calibrated multi-horizon gated recurrent unit (RC-MH-GRU) offline at 100, 200, 300, and 500 ms. The aggregate dataset contains 72 videos and 2,986 head-tracking sessions sampled at 10 Hz. Each two-degree-of-freedom viewing direction is mapped to a three-dimensional unit vector; one second of position and first-difference history forms the input. Ridge-AR and the 14,604-parameter MH-GRU are combined on the unit sphere with one validation-selected weight per horizon, yielding 15,336 parameters. Evaluation covers five training seeds, a video-disjoint test split, five-fold video cross-validation, strong baselines, ablation, computational profiling, video-level statistics, and a *parameter-matched* 14,412-parameter *Tiny Transformer*. Across 60,000 test windows, mean angular errors are 0.958°, 2.033°, 3.324°, and 6.041°, lower than Ridge-AR at all horizons and compared with 1.026°, 2.090°, 3.383°, and 6.118° for the *Tiny Transformer*. Cross-validation on 72 videos also favors RC-MH-GRU over Ridge-AR at 200–500 ms. The evidence supports a compact prediction component for prospective tile/ROI allocation, not bandwidth savings, viewport-quality gains, or end-to-end telepresence performance.

Kata Kunci:

Arah pandang,
Jaringan saraf berulang,
Prediksi multi-horizon,
Prediksi viewport,
Video 360 derajat,

ABSTRAK

Prediksi arah pandang dapat membantu sistem video 360 derajat menentukan bagian panorama yang perlu diprioritaskan pada streaming berbasis tile, region of interest (ROI), atau kualitas adaptif. Penelitian ini mengevaluasi Ridge-calibrated multi-horizon gated recurrent unit (RC-MH-GRU) secara offline pada horizon 100, 200, 300, dan 500 ms. Data agregat mencakup 72 video dan 2.986 sesi pelacakan kepala pada 10 Hz. Arah pandang 2DoF direpresentasikan sebagai vektor satuan tiga dimensi; riwayat posisi dan beda pertama selama satu detik menjadi masukan. Ridge-AR dan MH-GRU berparameter 14.604 dipadukan pada bola satuan dengan satu bobot setiap horizon yang dipilih melalui validasi, sehingga totalnya 15.336 parameter. Evaluasi mencakup lima seed, test split terpisah pada tingkat video, five-fold cross-validation, baseline kuat, ablasi, profil komputasi, statistik tingkat video, dan *Tiny Transformer* 14.412 parameter. Pada 60.000 window uji, rerata galat RC-MH-GRU adalah 0,958°, 2,033°, 3,324°, dan 6,041°, sedangkan *Tiny Transformer* memperoleh 1,026°, 2,090°, 3,383°, dan 6,118°. Cross-validation pada 72 video juga lebih baik daripada Ridge-AR pada 200–500 ms. Hasil mendukung komponen prediksi ringkas untuk calon alokasi tile/ROI, bukan bukti penghematan bandwidth, peningkatan kualitas viewport, atau kinerja telepresensi end-to-end.

Korespondensi Penulis:

Abdullah Iskandar,
Program Studi Informatika Universitas Telkom, Surabaya,
Jawa Timur 60231
Telepon : +6285186698276
Email: *abdullahiskandar@telkomuniversity.ac.id*

Submitted: 13-08-2026; Accepted: 14-10-2026;
Published: 16-10-2026

Copyright (c) 2026 The Author (s) This article is distributed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0)

1. PENDAHULUAN

Video 360 derajat merekam lingkungan sebagai panorama penuh, sedangkan pengguna mengamati bagian terbatas yang disebut *viewport*. Jika pemutar menerima dan mendekode seluruh panorama dengan kualitas seragam, pergeseran *viewport* melalui mouse berlangsung secara lokal. Dalam konfigurasi tersebut, prediksi arah pandang tidak memperbarui bingkai video yang telah terlambat dan tidak dengan sendirinya menurunkan latensi jaringan. Pembedaan fungsi ini penting agar prediksi tidak disamakan dengan mekanisme transmisi. Teleoperasi [1], [2], kunjungan jarak jauh [3], *embodied telepresence* [4], dan antarmuka manusia-mesin berlatensi rendah [5] tetap menjadi konteks pemanfaatan yang relevan. Namun, manfaat prediksi baru muncul ketika jalur media dapat memprioritaskan sebagian panorama.

Prioritas panorama dapat diterapkan melalui *streaming* berbasis *tile*, *region of interest* (ROI), kualitas adaptif, atau *decoding* parsial. Panorama dasar tetap dikirim untuk menjaga cakupan 360 derajat, sedangkan mutu lebih tinggi dialokasikan pada area yang diperkirakan akan menjadi *viewport* beberapa ratus milidetik kemudian. Jadi, prediksi tidak mengubah usia bingkai; model menyediakan waktu persiapan untuk memilih *tile*, *bitrate*, atau prioritas *decoding*. Kajian terkini merangkum pergeseran dari model gerak sederhana menuju jaringan berulang, mekanisme perhatian, dan *Transformer* [6]. Illahi dkk. mengkaji prediksi arah kepala untuk *remote rendering* [7], sedangkan Rondon dkk. memperlihatkan bahwa arsitektur kompleks tidak selalu mengungguli pembanding berbasis riwayat yang dikonfigurasi dengan baik [8].

Penelitian mutakhir mencakup estimasi *viewport* yang efisien [9], perbandingan teknik prediksi [10], fusi spasial-temporal untuk horizon panjang [11], serta optimasi bersama antara prediksi dan transmisi [12]. Arah lain menggunakan model difusi [13], personalisasi terfederasi [14], dan pengaliran adaptif ketika *viewport* tidak pasti [15], [16]. Keragaman tersebut memperlihatkan bahwa prediktor sederhana dan jaringan berulang dapat membawa informasi yang saling melengkapi. Kebutuhan yang masih relevan ialah keluaran beberapa horizon, evaluasi terhadap pembanding kuat, serta mekanisme yang memilih peran kedua sumber prediksi tanpa melihat data uji.

Atas dasar tersebut, penelitian ini mengusulkan *RC-MH-GRU* untuk memprediksi arah pandang pada horizon 100, 200, 300, dan 500 ms. *RC-MH-GRU* memadukan *Ridge-AR*, yang kuat untuk pola gerak lokal, dan *MH-GRU* bersama, yang mempelajari pola nonlinier dari riwayat sepuluh sampel. Kebaruannya bukan sel *GRU* baru, melainkan fusi pada geometri bola dengan bobot terpisah untuk setiap horizon yang dipilih hanya melalui validasi. Kontribusi penelitian meliputi: (1) representasi arah 2DoF sebagai vektor satuan tiga dimensi dengan beda pertama; (2) fusi *Ridge-AR* dan satu *MH-GRU* 14.604 parameter yang menghasilkan empat horizon dalam satu pemanggilan; (3) pemilihan bobot tanpa penggunaan data uji; serta (4) evaluasi video-disjoint dengan lima seed, five-fold cross-validation, statistik tingkat video, ablasi, dan profil komputasi. Pembanding mencakup *Persistence*, *Linear Extrapolation*, *Kalman Filter*, *Ridge-AR*, empat LSTM terpisah, TCN, dan *Tiny Transformer* berukuran sebanding. Klaim penerapan dibatasi pada prediksi arah sebagai komponen calon alokasi *tile*/ROI. Evaluasi penelitian ini dilakukan secara *offline* pada data arah kepala pengguna yang menonton video 360 derajat secara pasif. Istilah telepresensi pada judul menunjukkan konteks penerapan prediktor, bukan jenis *dataset* atau bukti bahwa sistem robot telah diuji secara *end-to-end*. Dalam rancangan penerapan, kamera 360 derajat di atas robot mengirim panorama kepada operator. Operator memilih *viewport* dengan *mouse*, sedangkan prediktor memperkirakan arah pandang berikutnya untuk membantu menyiapkan *tile*/ROI. Karena robot, jaringan, dan pengalaman operator belum diuji sebagai satu sistem, hasil penelitian dibatasi pada kinerja komponen prediksi.

2. METODE PENELITIAN

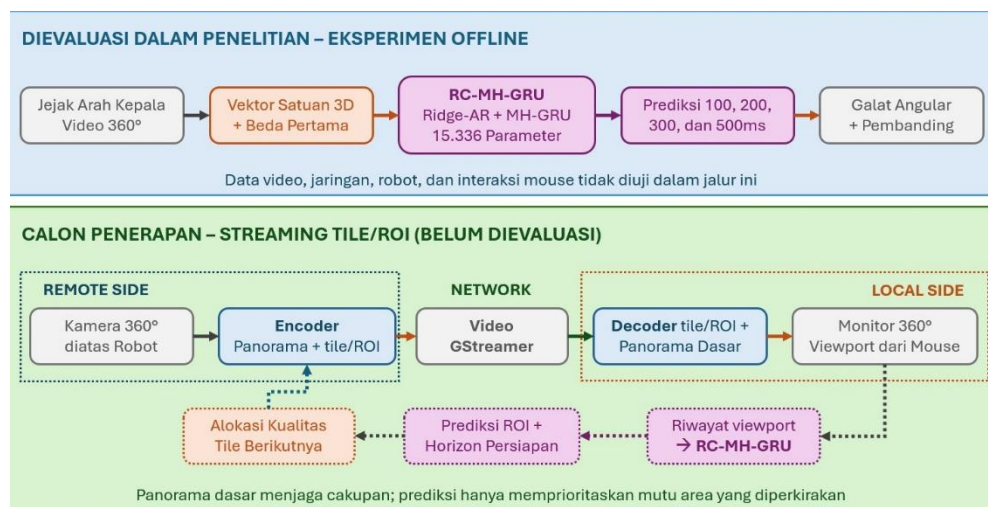
2.1 Ruang Lingkup dan Skenario Pemanfaatan

Penelitian ini mengevaluasi prediktor secara *offline* menggunakan *dataset* yang merekam arah kepala pengguna saat menonton video 360 derajat secara pasif. Pada pengamatan pasif, pengguna melihat konten yang telah direkam dan mengubah arah pandang tanpa mengendalikan pergerakan robot atau menjalankan tugas pencarian. Tindakan pengguna tidak mengubah posisi kamera maupun kondisi lingkungan. Teleoperasi aktif memiliki alur yang berbeda. Operator menggerakkan robot melalui keyboard, mencari objek, menghindari hambatan, dan mengubah *viewport* melalui *mouse* sambil menerima video serta telemetri sebagai umpan balik. Tujuan tugas dan umpan balik tersebut dapat menghasilkan pola arah pandang yang berbeda. Oleh karena itu, data pasif digunakan untuk menguji komponen prediktor secara terkendali, bukan untuk mewakili seluruh perilaku operator telepresensi. Hasil eksperimen menunjukkan kinerja prediksi pada data pasif dan belum membuktikan kinerja pada teleoperasi aktif. Model tidak menerima piksel video, data jaringan, perintah kendali robot, atau masukan mouse. Karena itu, hasil yang dapat dinilai langsung terbatas pada galat arah, kestabilan antar-pelatihan, generalisasi lintas video, jumlah parameter, dan waktu fungsi prediksi. Sebagai konteks arsitektur, PITON memisahkan kanal media dari komunikasi kendali [17]; prinsip pemisahan tersebut hanya menjadi rujukan bagi calon penerapan dan tidak diuji dalam eksperimen ini.

Gambar 1 membedakan jalur yang benar-benar dievaluasi dari calon jalur penerapan. Pada eksperimen *offline*, jejak arah kepala diubah menjadi vektor satuan dan beda pertama. *Ridge-AR* serta *MH-GRU* menghasilkan

prediksi empat horizon, kemudian keduanya dipadukan pada bola satuan menggunakan bobot hasil validasi. Output dibandingkan dengan *ground truth* dan *baseline*. Pada calon penerapan, kamera 360 derajat di sisi robot menghasilkan panorama. *Encoder* dapat mengirim panorama dasar dan meningkatkan kualitas *tile/ROI* yang diperkirakan akan dilihat. Sisi lokal mendekode panorama, menampilkan viewport, dan mencatat riwayat pergeseran mouse sebagai masukan prediktor. Alur terakhir masih berupa desain konseptual; penelitian tidak mengimplementasikan *GStreamer*, *WebSocket*, *tile encoding*, atau perangkat robot.

Kamera 360 derajat di bagian atas robot mengurangi risiko area penting berada di luar arah kamera, sebab panorama dasar tetap dapat menyediakan cakupan ke segala arah tanpa mekanisme *pan/tilt*. Karakteristik ini dapat membatasi dampak prediksi yang keliru: sistem masih menampilkan versi dasar area di luar *ROI*, meskipun mutunya mungkin lebih rendah. Perlindungan tersebut bukan tanpa biaya. Panorama penuh membutuhkan bitrate dan decoding yang lebih besar, sementara gerakan platform dapat menimbulkan *optical flow*, getaran, artefak stitching, oklusi nadir, serta perubahan pencahayaan. Dengan demikian, kamera 360 derajat menutup sebagian kelemahan kamera berarah, tetapi tidak menghapus kebutuhan pengujian kualitas layanan dan pengalaman pengguna pada sistem nyata.



Gambar 1. Pemisahan ruang lingkup evaluasi *offline* dan calon penerapan *RC-MH-GRU* pada *streaming* berbasis *tile* atau *ROI*.

2.2 Dataset dan Pembagian Data

Dataset menggabungkan enam studi pelacakan kepala saat pengguna menonton video 360 derajat [18], [19], [20], [21], [22], [23]. Penelitian ini tidak menggunakan berkas video maupun frame video sebagai masukan. Data yang dianalisis adalah jejak arah kepala yang telah diseragamkan oleh penyedia dataset menjadi 10 Hz; dengan demikian, *FPS* video sumber bukan variabel penelitian. Dokumentasi dataset mencantumkan 88 ID video. Hasil pemeriksaan data menunjukkan bahwa jejak arah kepala untuk ID 17–32 tidak tersedia pada salinan yang digunakan. Program tidak melakukan imputasi terhadap data yang tidak tersedia dan hanya memproses berkas yang dapat diparsing, sehingga diperoleh 72 data video dengan 2.986 sesi pengguna-video. Keterbatasan lain terdapat pada indeks pengguna yang dimulai kembali pada setiap video, sehingga identitas global tidak dapat ditelusuri lintas konten. Oleh karena itu, evaluasi ini mengukur generalisasi ke video yang belum terlihat, bukan ke pengguna baru.

Semua trayektori telah diseragamkan pada 10 Hz dan menyimpan elevasi serta azimuth dalam radian. *Split* utama dibentuk pada tingkat video *seed* 42. Dirincikan pada Tabel 1 dengan 50 video untuk pelatihan, 11 untuk validasi, dan 11 untuk pengujian. Setelah pembentukan *window*, subsampel acak tanpa penggantian mempertahankan 200.000, 40.000, dan 60.000 *window*. Konten uji tidak pernah masuk ke pelatihan. Satu pembagian data berpotensi menghasilkan estimasi kinerja yang bergantung pada komposisi sampel, sehingga *five-fold cross-validation* juga diterapkan. Selain pembagian utama pada Tabel 1, *five-fold cross-validation* dilakukan secara terpisah pada seluruh 72 video. Setiap video menjadi data uji tepat satu kali; kelima fold uji masing-masing memuat 15, 15, 14, 14, dan 14 video. Pada setiap fold, dari video yang tersisa, 9 video digunakan untuk validasi dan 49 video digunakan untuk pelatihan.

Tabel 1. Pembagian utama berdasarkan hasil eksekusi ulang.

Bagian	Video	Sesi	<i>Window</i> terbentuk	<i>Window</i> dipakai
Latih	50	2.065	1.071.142	200.000
Validasi	11	432	248.932	40.000
Uji	11	489	282.468	60.000
Total	72	2.986	1.602.542	300.000

2.3 Representasi Arah dan *Windowing*

Istilah “pose kepala 3DoF” sengaja tidak digunakan. Dataset hanya mencatat dua sudut rotasi, sehingga variabel targetnya adalah arah pandang 2DoF. Vektor satuan tiga dimensi berfungsi sebagai pemetaan geometris, bukan derajat kebebasan tambahan. Bentuk ini menjaga azimuth tetap kontinu saat melintasi $-\pi$ dan π . Dengan elevasi ϕ_t serta azimuth θ_t , vektor arah $\mathbf{u}_t$ ditulis pada Persamaan (1).

$$\mathbf{u}_t = [\cos\phi_t \cos\theta_t, \cos\phi_t \sin\theta_t, \sin\phi_t]^T \quad (1)$$

Sebagai contoh, model menggunakan riwayat sepuluh sampel—setara satu detik—dengan *stride* tiga sampel. Target diletakkan satu, dua, tiga, dan lima langkah setelah pengamatan terakhir. Pada laju 10 Hz, susunan itu tepat bersesuaian dengan horizon 100, 200, 300, dan 500 ms. Setiap langkah memuat $\mathbf{u}_t$ serta beda pertama $\Delta\mathbf{u}_t$ seperti Persamaan (2). Naskah tidak menyebut data model 90 Hz karena tidak ada pengambilan maupun interpolasi *label* pada laju tersebut.

$$\Delta\mathbf{u}_t = \mathbf{u}_t - \mathbf{u}_{t-1}, \quad \mathbf{X}_t = [\mathbf{u}_t, \Delta\mathbf{u}_t] \quad (2)$$

2.4 Model dan Pelatihan

MH-GRU terdiri atas satu lapisan GRU dengan 64 hidden units [24]. Hidden state terakhir masuk ke linear layer berisi 12 output, lalu dibentuk menjadi empat vektor tiga dimensi dan dinormalisasi ke bola satuan. Model ini memiliki 14.604 parameter dan menghasilkan empat horizon dalam satu pemanggilan. Cosine loss pada Persamaan (3) digunakan sebagai fungsi objektif, sedangkan Adam [25] menjalankan optimisasi dengan learning rate 10^{-3} dan batch size 1.024. Pelatihan dibatasi 20 epoch; early stopping dengan patience empat mengikuti rerata galat angular validasi.

Pembandingan disusun dari metode sederhana hingga model pembelajaran. Persistence hanya menyalin arah terakhir. Linear Extrapolation memasang garis kuadrat terkecil pada sepuluh sampel setiap komponen, sedangkan Filter Kalman menggunakan keadaan posisi-kecepatan konstan dengan $Q=10^{-3}$ dan $R=10^{-2}$. Ridge-AR memetakan 60 fitur riwayat ke 12 output menggunakan regularisasi Ridge [26]; data validasi memilih λ dari $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10\}$. Empat LSTM terpisah [27] mewakili model proyek sebelumnya dengan 74.508 parameter, sementara TCN multi-horizon dua lapis konvolusi satu dimensi 64 kanal menyediakan arsitektur nonberulang dengan 14.348 parameter. Pembandingan modern menggunakan *Tiny Transformer* satu lapis dengan $d_{\text{model}}=40$, empat *attention heads*, dimensi *feed-forward* 80, *positional embedding* terpelajar, dan *linear head* 12 output [28]. Konfigurasi 14.412 parameter ini menghasilkan empat horizon sekaligus sehingga anggaran parameternya sebanding dengan MH-GRU dan RC-MH-GRU.

Ablasi dilakukan dengan menghapus beda pertama atau mengganti objektif kosinus menjadi galat kuadrat rerata. MH-GRU lengkap dan *Tiny Transformer* dilatih lima kali menggunakan *seed* 42–46. *Tiny Transformer* menerima masukan 10×6 yang sama, menghasilkan empat vektor keluaran, menggunakan objektif kosinus, *batch size* 1.024, dan *early stopping* berdasarkan data validasi. Untuk RC-MH-GRU, Ridge-AR selalu dipasang hanya pada data latih; λ dan empat bobot fusi dipilih pada data validasi. Untuk tiap horizon h , kedua keluaran vektor satuan dipadukan sesuai Persamaan (4). Pemilihan model, λ , dan bobot selesai sebelum *test split* digunakan. Pada *cross-validation*, prosedur Ridge-AR, pelatihan MH-GRU, dan pemilihan bobot diulang pada setiap *fold*. Jumlah *window* per video diseimbangkan agar video panjang tidak mendominasi kontribusi terhadap metrik evaluasi.

$$L = \frac{1}{4N} \sum_i \sum_h [1 - \hat{\mathbf{u}}_{i,h} \cdot \mathbf{u}_{i,h}] \quad (3)$$

$$\hat{\mathbf{u}}_{i,h}^{\text{fus}} = \text{normalize}[(1 - \alpha_h) \hat{\mathbf{u}}_{i,h}^{\text{Ridge}} + \alpha_h \hat{\mathbf{u}}_{i,h}^{\text{MH-GRU}}], \quad \alpha_h \in [0, 1] \quad (4)$$

2.5 Metrik dan Analisis Statistik

Pada Persamaan (4), α_h adalah bobot MH-GRU pada horizon h . Bobot dicari pada *grid* 0,00–1,00 dengan langkah 0,01 menggunakan rerata galat angular validasi, lalu dikunci. Persamaan (5) menghitung galat angular e antara prediksi $\hat{\mathbf{u}}$ dan *ground truth* $\mathbf{u}$. Dari 60.000 *window* uji, evaluasi melaporkan rerata, median, akar galat kuadrat rerata (RMSE), serta persentil ke-90 (P90). Simpangan baku antar-*seed* berasal dari lima pelatihan MH-GRU sebelum fusi.

Window yang saling tumpang tindih tidak diperlakukan sebagai observasi independen. Galat mula-mula dirata-ratakan per video pada 11 video uji, baru kemudian dibandingkan secara berpasangan dengan uji Wilcoxon eksak. Selang kepercayaan 95% untuk selisih rerata diperoleh lewat 10.000 *bootstrap* tingkat video. Evaluasi utama memuat 16 perbandingan—Persistence, Linear Extrapolation, Kalman Filter, dan *Tiny Transformer* terhadap RC-MH-GRU pada empat horizon—sehingga koreksi Bonferroni menetapkan ambang 0,003125. Untuk *cross-validation*, rerata makro dihitung atas seluruh 72 video dan selang kepercayaannya menggunakan prosedur *bootstrap* yang sama.

Profil komputasi dijalankan dengan *batch size* satu pada CPU Apple M3 delapan inti, memori 16 GB, dan satu utas PyTorch. Setelah *warm-up*, setiap metode menjalani 3.000 *inference calls*. Median, P95, dan P99 kemudian

dicatat. Pengukuran tersebut hanya mencakup fungsi prediksi; kamera, jaringan, *decoding*, *rendering*, dan layar berada di luar pengukuran.

$$e(\hat{\mathbf{u}}, \mathbf{u}) = \frac{180}{\pi} \arccos[\text{clip}(\hat{\mathbf{u}} \cdot \mathbf{u}, -1, 1)] \quad (5)$$

3. HASIL DAN PEMBAHASAN

3.1 Akurasi dan Keseimbangan *Baseline*

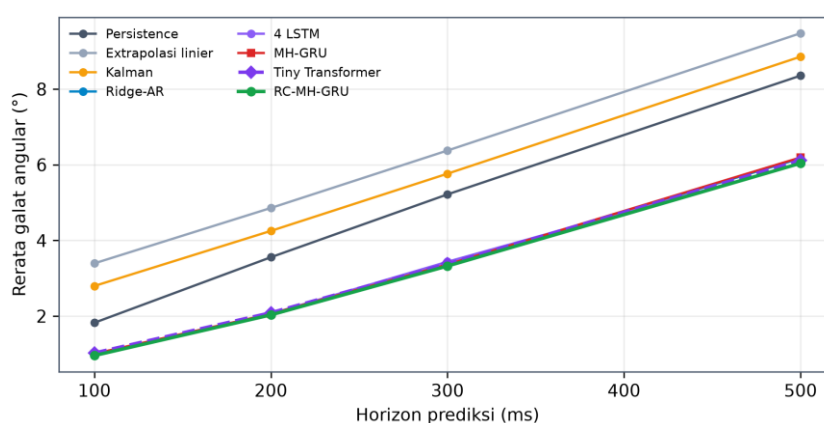
Eksekusi ulang berhasil membaca 72 video dan 2.986 sesi. Jumlah *window* yang dihasilkan juga sama dengan metadata program. Pada Tabel 2, *RC-MH-GRU* menurunkan rerata galat *Persistence* sebesar 47,5%, 42,8%, 36,3%, dan 27,7% untuk horizon 100–500 ms. Keunggulan absolut terhadap *Persistence* meningkat dari 0,867° menjadi 2,317° ketika horizon memanjang. Artinya, komponen *Ridge-AR* menangkap kecenderungan lokal yang kuat, sedangkan *MH-GRU* memberi koreksi nonlinier yang berguna saat target makin jauh.

Tabel 2 menunjukkan bahwa *RC-MH-GRU* memperoleh rerata galat *per-window* terendah pada empat horizon: 0,958°, 2,033°, 3,324°, dan 6,041°. Nilai tersebut lebih rendah daripada *Ridge-AR* pada seluruh horizon serta lebih rendah daripada empat *LSTM* terpisah pada 100–300 ms; pada 500 ms nilainya 0,004° lebih tinggi daripada *LSTM*. Rerata empat horizon *RC-MH-GRU* adalah 3,089°, dibandingkan 3,140° untuk *Ridge-AR*, 3,135° untuk empat *LSTM*, dan 3,157° untuk *MH-GRU*. Karena selisih terhadap *baseline* kuat kecil dan *window* bertumpang tindih, hasil tersebut diperlakukan sebagai bukti deskriptif. Klaim statistik tetap ditentukan dari rerata tingkat video dan *cross-validation*. *Tiny Transformer* menghasilkan rerata 1,026°, 2,090°, 3,383°, dan 6,118°, sehingga *RC-MH-GRU* lebih rendah pada seluruh horizon dengan selisih masing-masing 0,068°, 0,057°, 0,059°, dan 0,077°.

Lima pelatihan *MH-GRU* menghasilkan variasi yang rendah. Untuk empat horizon, simpangan baku rerata galat antar-seed sebelum fusi berturut-turut hanya 0,029°, 0,023°, 0,048°, dan 0,037°. Sesudah *Ridge-AR* dipadukan menggunakan bobot validasi yang dipilih terpisah pada setiap *seed*, rerata *RC-MH-GRU* tetap berada di bawah *Ridge-AR* pada keempat horizon *test split*. Kestabilan inisialisasi ini belum menjawab pergantian konten video; evaluasi generalisasi lintas video dilakukan melalui *cross-validation*. Perbandingan ini juga dapat terlihat pada Gambar 2. *Tiny Transformer* juga dievaluasi dengan lima *seed*; simpangan baku rerata galatnya pada horizon 100–500 ms adalah 0,031°, 0,022°, 0,039°, dan 0,049°.

Tabel 2. Rerata galat angular pada 60.000 window uji (derajat)

Horizon (ms)	Persistence (°)	Linear Extrapol. (°)	Kalman Filter (°)	Ridge-AR (°)	4 LSTM (°)	RC-MH-GRU (°)	<i>Tiny Transformer</i> (°)
100	1,825	3,395	2,797	0,963	1,002	0,958	1,026
200	3,557	4,859	4,253	2,059	2,063	2,033	2,090
300	5,218	6,377	5,765	3,368	3,437	3,324	3,383
500	8,357	9,477	8,858	6,169	6,037	6,041	6,118



Gambar 2. Rerata galat angular delapan metode terhadap horizon prediksi.

3.2 Statistik Tingkat Video dan *Cross-Validation*

Pada 11 video uji, *RC-MH-GRU* tetap menghasilkan galat yang lebih rendah daripada *Persistence*, *Linear Extrapolation*, dan *Kalman Filter* pada setiap video dan horizon; pola tersebut mempertahankan seluruh 12 hasil Wilcoxon eksak $p=0,0009766$ setelah koreksi Bonferroni. Statistik berikut menggunakan galat yang dirata-ratakan atas lima pelatihan independen, konsisten dengan Tabel 3. Perbandingan terhadap *baseline* kuat perlu dibaca lebih hati-hati. Pada *test split*, *RC-MH-GRU* lebih rendah daripada *Ridge-AR* pada 7/11 video untuk 100 ms ($p=0,1230$) dan

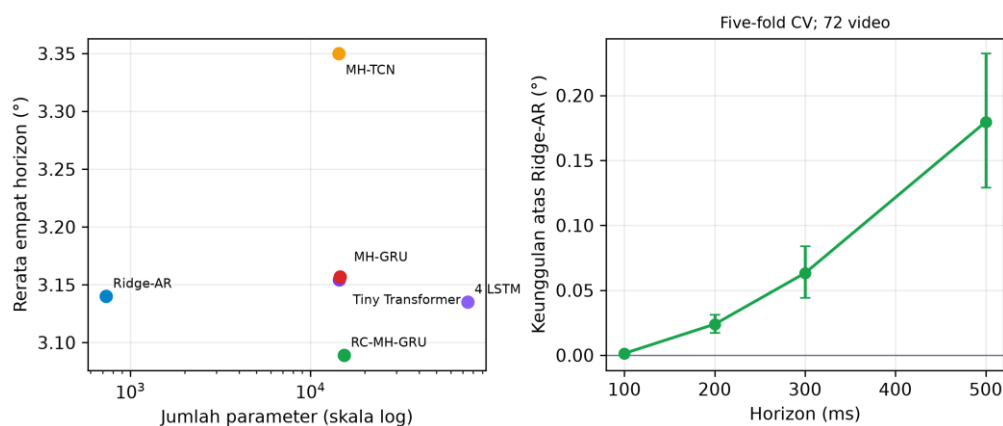
pada 6/11 video untuk tiap horizon lain; tidak satu pun hasil melewati ambang terkoreksi 0,00417. Terhadap empat *LSTM*, keunggulan pada 100 ms signifikan ($p=0,0009766$), tetapi hasil 200–500 ms tidak melewati ambang terkoreksi. Dengan demikian, data tidak mendukung klaim keunggulan statistik *RC-MH-GRU* atas setiap *baseline* kuat pada *test split*.

Perbandingan langsung dengan *Tiny Transformer* pada 11 video uji menggunakan rerata lima seed menunjukkan 100 ms: $\Delta=-0,083^\circ$, $p=0,00098$, *RC-MH-GRU* unggul pada 11/11 video; 200 ms: $\Delta=-0,061^\circ$, $p=0,00098$, *RC-MH-GRU* unggul pada 11/11 video; 300 ms: $\Delta=-0,063^\circ$, $p=0,00293$, *RC-MH-GRU* unggul pada 10/11 video; 500 ms: $\Delta=-0,068^\circ$, $p=0,02441$, *RC-MH-GRU* unggul pada 10/11 video. Tanda negatif pada $\Delta=\text{RC-MH-GRU}-\text{Tiny Transformer}$ berarti galat model usulan lebih rendah. Hasil ini mendukung keunggulan pada pembagian data tes utama, tetapi ukuran 11 video dan koreksi Bonferroni 0,003125 tetap membatasi klaim signifikansi.

Five-fold cross-validation memperluas pemeriksaan ke seluruh 72 video. Grafiknya dapat dilihat pada Gambar 3. Rerata makro *RC-MH-GRU* mencapai 1,168, 2,519, 4,145, dan 7,586, sedangkan *Ridge-AR* mencatat 1,169, 2,543, 4,208, dan 7,766. Selisih *RC-MH-GRU-Ridge-AR* adalah $-0,001^\circ$ [$IK_{95\%} -0,003; -0,000$], $-0,024^\circ$ [$-0,031; -0,017$], $-0,063^\circ$ [$-0,084; -0,044$], dan $-0,180^\circ$ [$-0,232; -0,129$]. Uji *Wilcoxon* menghasilkan $p=0,00547$, $1,45 \times 10^{-8}$, $3,05 \times 10^{-8}$, dan $1,92 \times 10^{-9}$. Dengan ambang konservatif 0,00417 yang digunakan untuk 12 perbandingan utama, bukti keunggulan lintas konten kuat pada 200–500 ms; $p=0,00547$ pada 100 ms tidak melewati ambang tersebut. *RC-MH-GRU* lebih rendah pada 47, 57, 54, dan 61 dari 72 video. Generalisasi lintas pengguna belum dapat dinilai karena identitas global pengguna tidak tersedia.

Tabel 3. *Five-fold cross-validation: RC-MH-GRU dibandingkan Ridge-AR.*

Horizon (ms)	RC-MH-GRU ($^\circ$)	Ridge-AR ($^\circ$)	Selisih ($^\circ$)	IK 95% selisih ($^\circ$)	RC-MH-GRU lebih rendah (n/72)
100	1,168	1,169	-0,001	$[-0,003; -0,000]$	47/72
200	2,519	2,543	-0,024	$[-0,031; -0,017]$	57/72
300	4,145	4,208	-0,063	$[-0,084; -0,044]$	54/72
500	7,586	7,766	-0,180	$[-0,232; -0,129]$	61/72



Gambar 3. Pertukaran jumlah parameter-akurasi dan keunggulan *RC-MH-GRU* atas *Ridge-AR* pada *five-fold cross-validation*.

3.3 Ablasi dan Profil Komputasi

Tabel 4 memperlihatkan peran beda pertama dan *cosine loss* pada cabang *MH-GRU*. Ketika beda pertama dihapus, rerata empat horizon *MH-GRU* naik dari 3,157 $^\circ$ menjadi 3,383 $^\circ$. Penggantian objektif kosinus dengan galat kuadrat juga menaikkannya menjadi 3,210 $^\circ$. *TCN* multi-horizon menghasilkan rerata 3,350 $^\circ$. Hasil satu *seed* tersebut tidak cukup untuk menyatakan *GRU* selalu unggul atas *TCN*; ablasi hanya mendukung kontribusi positif kedua pilihan desain pada konfigurasi yang diuji. Fusi *RC-MH-GRU* tidak menghapus temuan ini, melainkan memanfaatkan keluaran *MH-GRU* bersama pembanding *Ridge-AR* yang telah disetel melalui validasi. *Tiny Transformer* memperoleh rerata 3,154 $^\circ$ dengan 14.412 parameter. Nilai ini lebih baik daripada pembanding pembelajaran tersebut, tetapi fusi validasi *RC-MH-GRU* menghasilkan rerata 3,089 $^\circ$ yang lebih rendah daripada pembanding pembelajaran tersebut.

Tabel 5 merincikan profil pada setiap satu sampel dengan 3.000 panggilan. Untuk satu sampel, median *inference call RC-MH-GRU* adalah 81,834 μs , dengan P95 83,500 μs dan P99 86,959 μs . Pada mesin yang sama, median *Persistence*, *Linear Extrapolation*, *Kalman Filter*, dan *MH-GRU* masing-masing mencapai 1,000 μs , 26,417 μs , 179,084 μs , dan 72,333 μs . Tambahan *Ridge-AR* menaikkan biaya *MH-GRU* sekitar 9,5 μs , tetapi totalnya tetap jauh di bawah interval sampling 100 ms. Pengukuran ini hanya mencakup fungsi prediksi; kamera, jaringan, decoding,

rendering, dan layar belum masuk ke pengukuran. Oleh sebab itu, hasil ini tidak mendukung klaim kinerja sistem telepresensi waktu nyata secara *end-to-end*. Pada pengukuran yang sama, *Tiny Transformer* mencapai median 69,500 μ s, P95 71,713 μ s, dan P99 80,005 μ s. Perbandingan ini hanya mencakup fungsi prediksi dan tidak membuktikan kinerja telepresensi waktu nyata secara *end-to-end*.

Tabel 4. Ablasi MH-GRU dan perbandingan arsitektur pada pembagian utama.

Konfigurasi	Rerata empat horizon (°)	Jumlah parameter
MH-GRU lengkap	3,157	14604
Tanpa beda pertama	3,383	14028
Objektif MSE	3,210	14604
MH-TCN	3,350	14348
Ridge-AR	3,140	732
Empat LSTM terpisah	3,135	74508
<i>Tiny Transformer</i>	3,154	14.412

Tabel 5. Profil waktu inferensi satu sampel pada 3.000 panggilan.

Metode	Median (μ s)	P95 (μ s)	P99 (μ s)
Persistence	1,000	1,084	1,208
Linear Extrapolation	26,417	26,834	29,875
Kalman Filter	179,084	182,752	188,795
MH-GRU	72,333	74,375	79,334
RC-MH-GRU	81,834	83,500	86,959
<i>Tiny Transformer</i>	69,500	71,713	80,005

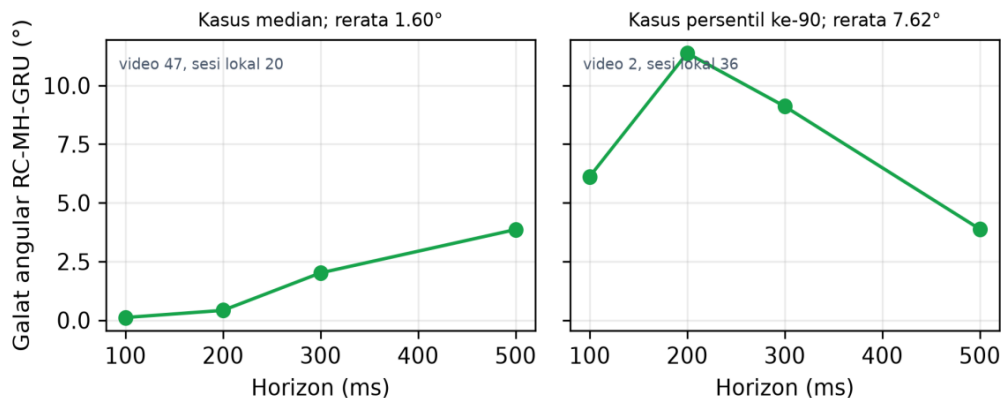
3.4 Implikasi untuk *Streaming Adaptif* dan Keterbatasan

Hasil penelitian memberikan tiga implikasi. Pertama, pembandingan yang terlalu lemah dapat membesarkan kebaruan. Ridge-AR hanya memiliki 732 parameter dan kinerjanya sangat kuat. Temuan ini konsisten dengan hasil Rondon dkk. [8] dan kajian perbandingan terbaru [10]: arsitektur kompleks harus diuji terhadap pembandingan autoregresif yang dikonfigurasi secara memadai. Kedua, fusi yang dipilih melalui validasi dapat memanfaatkan kekuatan komplementer tersebut: Ridge-AR memodelkan kecenderungan lokal, sedangkan MH-GRU memberi koreksi nonlinier untuk beberapa horizon. Sebagai ilustrasi konseptual pada calon teleoperasi, operator dapat menggeser mouse ke kanan untuk memindahkan viewport. Ridge-AR cenderung meneruskan arah dan kecepatan gerak terakhir, sedangkan MH-GRU dapat mengoreksinya ketika gerakan mulai melambat atau segera berubah arah berdasarkan pola riwayat satu detik. RC-MH-GRU memadukan kedua keluaran tersebut. Skenario mouse ini menjelaskan penggunaan model, bukan hasil eksperimen, karena data penelitian berasal dari pelacakan kepala. Ketiga, RC-MH-GRU tetap ringkas pada 15.336 parameter—79,4% lebih sedikit daripada empat LSTM terpisah—dan menghasilkan empat horizon dalam satu panggilan. Kontribusi yang didukung data adalah fusi multi-horizon yang tervalidasi serta evaluasi transparan, bukan dominasi statistik tanpa syarat atas semua metode. *Tiny Transformer* menguji apakah mekanisme perhatian berukuran sebanding dapat menggantikan kombinasi tersebut. RC-MH-GRU menghasilkan galat lebih rendah pada empat horizon, tetapi *Tiny Transformer* sedikit lebih kecil dan lebih cepat pada profil CPU. Hasil ini menunjukkan pertukaran antara akurasi arah dan biaya komputasi, bukan dominasi universal satu arsitektur.

Dalam calon *streaming tile/ROI*, empat horizon dapat berfungsi sebagai pilihan waktu persiapan. Misalkan D_{prep} menyatakan interval yang diperlukan sejak prediksi dibuat hingga tile berkualitas tinggi siap dirender, termasuk permintaan ROI, pengodean, transmisi, dan decoding. Persamaan (6) memilih horizon tersedia yang paling dekat dengan interval tersebut. Jika D_{prep} diperkirakan 180 ms, sistem memilih 200 ms; jika meningkat menjadi 430 ms, sistem memilih 500 ms. Contoh ini hanya menjelaskan cara memakai keluaran *RC-MH-GRU*. Nilai D_{prep} tidak diukur dalam penelitian, sehingga Persamaan (6) tidak boleh ditafsirkan sebagai bukti kompensasi latensi atau kinerja waktu nyata.

$$h^* = \arg \min_{h \in \{100, 200, 300, 500\}} |h - D_{\text{prep}}| \quad (6)$$

Pada Persamaan (6), D_{prep} adalah estimasi interval persiapan *tile/ROI* dalam milidetik; nilainya belum diukur pada sistem *end-to-end*.



Gambar 4. Contoh median dan P90 RC-MH-GRU yang dipilih secara algoritmik dari data uji.

Gambar 4 memperlihatkan dua *window* yang dipilih secara algoritmik berdasarkan kriteria statistik, bukan pemilihan subjektif: satu paling dekat dengan median dan satu lagi dengan P90 untuk *seed* 42. Kasus median bertahan di bawah 3° pada semua horizon. Pada kasus P90, galat meningkat hingga mendekati 20° pada horizon 500 ms. Ekor distribusi ini penting untuk calon alokasi ROI karena kesalahan besar dapat menempatkan mutu tinggi pada *tile* yang keliru. Panorama dasar 360 derajat dapat menjaga visibilitas area tersebut, tetapi manfaat dan penurunan mutunya baru dapat dinilai melalui simulasi *streaming* atau uji pengguna.

Penelitian ini memiliki beberapa batasan. Dataset merekam arah kepala saat menonton video 360 derajat secara pasif, bukan riwayat *viewport* dari mouse ketika operator mengendalikan robot. Laju 10 Hz juga lebih rendah daripada laju render layar, walaupun horizon target tetap konsisten dengan interval sampel. Model tidak menggunakan piksel, konteks adegan, gerak robot, ataupun ketidakpastian jaringan. Selain itu, *GStreamer*, *WebSocket*, *tile/ROI encoding*, penghematan bandwidth, kualitas *viewport*, keselamatan kendali, dan latensi *end-to-end* belum diuji. Profil komputasi hanya mengukur fungsi prediksi pada satu komputer. Keunggulan *RC-MH-GRU* atas *Ridge-AR* paling kuat pada *cross-validation* 200–500 ms; pada *test split* 11 video, beberapa selang kepercayaan tetap melintasi nol. Perbandingan dengan arsitektur pembelajaran mendalam yang lebih baru, seperti *Transformer*, juga belum merupakan klaim kompetitif penuh. Penelitian berikutnya perlu menggunakan riwayat mouse pada tugas teleoperasi nyata, mengukur manfaat *tile/ROI end-to-end*, dan menilai kualitas *viewport* serta keselamatan operator. Dengan demikian, hubungan penelitian ini dengan telepresensi berada pada tingkat rancangan penerapan. Model diuji sebagai calon komponen prediksi, sedangkan robot, alur *streaming*, dan interaksi operator secara *end-to-end* belum dievaluasi. Istilah telepresensi pada judul tidak dimaksudkan sebagai klaim bahwa *dataset* berasal dari teleoperasi atau bahwa kinerja sistem telepresensi telah terbukti. Pembanding modern dibatasi pada satu *Tiny Transformer parameter-matched*; arsitektur *Transformer* lebih besar, model berbasis piksel atau konteks adegan, personalisasi, serta model difusi tidak diuji langsung. Karena itu, hasil ini tidak boleh digeneralisasi menjadi penolakan terhadap seluruh arsitektur perhatian atau model difusi.

4. KESIMPULAN

Penelitian ini mengevaluasi *RC-MH-GRU* untuk prediksi arah pandang 2DoF pada video 360 derajat. Model memadukan *Ridge-AR* dan satu *MH-GRU* 14.604 parameter, lalu menghasilkan empat horizon—100, 200, 300, dan 500 ms—dalam satu pemanggilan. Pada 60.000 *window test split*, rerata galatnya mencapai $0,958^\circ$ – $6,041^\circ$ dan 27,7–47,5% lebih rendah daripada *Persistence*. Galat *per-window* tersebut juga lebih rendah daripada *Ridge-AR* dan *Tiny Transformer* pada keempat horizon. *Five-fold cross-validation* lintas 72 video memperkuat keunggulan atas *Ridge-AR* pada horizon 200–500 ms dengan ambang konservatif 0,003125. Pada *test split* 11 video, keunggulan atas *baseline* kuat tidak signifikan di seluruh horizon; karena itu, penelitian tidak mengklaim dominasi statistik universal. Kontribusi utama adalah fusi multi-horizon ringkas yang dipilih melalui validasi, disertai evaluasi terbuka terhadap pembanding kuat. Pengujian pada riwayat mouse, *streaming tile/ROI*, dan sistem telepresensi *end-to-end* tetap diperlukan sebelum manfaat sistem dapat dinyatakan. Pada *test split* 11 video, perbedaan terhadap *Tiny Transformer* memenuhi ambang terkoreksi pada 100–300 ms, tetapi tidak pada 500 ms; hasil ini tidak mendukung klaim dominasi statistik universal.

KETERSEDIAAN DATA DAN KODE

Kode, konfigurasi, pembagian data, skrip analisis statistik, dan keluaran eksperimen tersedia dalam paket reproduksi yang disiapkan bersama naskah. Repositori publik: https://github.com/Abdis-Dev/RC-MH-GRU_PrediksiArahPandang. *Dataset* pelacakan kepala bersumber dari A Large *Dataset* of 360 Video User Behaviour dan diperoleh melalui <https://github.com/360VidStr/A-large-dataset-of-360-video-user-behaviour>. Data mentah tidak disertakan dalam paket karena distribusinya mengikuti ketentuan sumber asli.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada para peneliti yang menyediakan dataset pelacakan kepala video 360 derajat secara terbuka sehingga evaluasi lintas video pada penelitian ini dapat dilakukan.

REFERENSI

- [1] S. B. Kamtam, Q. Lu, F. Bouali, O. C. L. Haas, and S. Birrell, "Network Latency in Teleoperation of Connected and Autonomous Vehicles: A Review of Trends, Challenges, and Mitigation Strategies," *Sensors*, vol. 24, no. 12, Jun. 2024, doi: 10.3390/s24123957.
- [2] A. Naceri *et al.*, "The Vicarios Virtual Reality Interface for Remote Robotic Teleoperation: Teleporting for Intuitive Tele-manipulation," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 101, no. 4, Apr. 2021, doi: 10.1007/s10846-021-01311-7.
- [3] R. Yahada and T. Ishida, "Implementation of a teleimmersion homecoming support system for supporting inpatients," *Internet of Things (Netherlands)*, vol. 17, Mar. 2022, doi: 10.1016/j.iot.2021.100491.
- [4] N. Arora *et al.*, "Augmenting Immersive Telepresence Experience with a Virtual Body," *IEEE Trans. Vis. Comput. Graph.*, vol. 28, no. 5, pp. 2135–2145, May 2022, doi: 10.1109/TVCG.2022.3150473.
- [5] J. Li, D. Han, J. Xu, K. Li, and Z. Ma, "Design and evaluation of Avatar: An ultra-low-latency immersive human-machine interface for teleoperation," *Displays*, vol. 92, Apr. 2026, doi: 10.1016/j.displa.2025.103292.
- [6] M. Z. A. Wahba, S. Baldoni, and F. Battisti, "Learning-Based Viewport Prediction for 360-Degree Videos: A Review," *Electronics (Switzerland)*, vol. 14, no. 18, Sep. 2025, doi: 10.3390/electronics14183743.
- [7] G. K. Illahi, A. Vaishnav, T. Kämäräinen, M. Siekkinen, and M. Di Francesco, "Learning to Predict Head Pose in Remotely-Rendered Virtual Reality," in *MMSys 2023 - Proceedings of the 14th ACM Multimedia Systems Conference*, Association for Computing Machinery, Inc, Jun. 2023, pp. 27–38. doi: 10.1145/3587819.3590972.
- [8] M. F. R. Rondon, L. Sassatelli, R. Aparicio-Pardo, and F. Precioso, "TRACK: A New Method from a Re-Examination of Deep Architectures for Head Motion Prediction in 360 Videos," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 5681–5699, Sep. 2022, doi: 10.1109/TPAMI.2021.3070520.
- [9] N. Viet Hung, T. Thanh Lam, T. Thanh Binh, A. Marshal, and T. Thu Huong, "Efficient Deep Learning-Based Viewport Estimation for 360-Degree Video Streaming," *Advances in Science, Technology and Engineering Systems Journal*, vol. 9, no. 3, pp. 49–61, May 2024, doi: 10.25046/aj090305.
- [10] M. Mahmoud *et al.*, "A Comparative Analysis of Viewing Prediction Techniques for 360° Video Streaming Applications," in *2024 Panhellenic Conference on Electronics and Telecommunications, PACET 2024 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/PACET60398.2024.10497019.
- [11] B. Gao *et al.*, "STAR-VP: Improving Long-term Viewport Prediction in 360° Videos via Space-aligned and Time-varying Fusion," in *MM 2024 - Proceedings of the 32nd ACM International Conference on Multimedia*, Association for Computing Machinery, Inc, Oct. 2024, pp. 5556–5565. doi: 10.1145/3664647.3681268.
- [12] M. Setayesh and V. W. S. Wong, "Viewport Prediction, Bitrate Selection, and Beamforming Design for THz-Enabled 360° Video Streaming," *IEEE Trans. Wirel. Commun.*, vol. 24, no. 3, pp. 1849–1865, 2025, doi: 10.1109/TWC.2024.3513221.
- [13] H. Zheng, L. Du, X. Nie, and F. Dong, "DiffVP: A Diffusion Model with Explicit Coordinate-Temporal Encoding for Viewport Prediction in 360° Videos," *Electronics (Switzerland)*, vol. 15, no. 6, Mar. 2026, doi: 10.3390/electronics15061326.
- [14] M. Setayesh and V. W. S. Wong, "A Content-based Viewport Prediction Framework for 360° Video Using Personalized Federated Learning and Fusion Techniques," in *Proceedings - IEEE International Conference on Multimedia and Expo*, IEEE Computer Society, 2023, pp. 654–659. doi: 10.1109/ICME55011.2023.00118.
- [15] L. Zhao, Y. Cui, Z. Liu, Y. Zhang, and S. Yang, "Adaptive Streaming of 360 Videos with Perfect, Imperfect, and Unknown FoV Viewing Probabilities in Wireless Networks," *IEEE Transactions on Image Processing*, vol. 30, pp. 7744–7759, 2021, doi: 10.1109/TIP.2021.3099741.
- [16] W. Feng, S. Wang, and Y. Dai, "Adaptive 360-Degree Streaming: Optimizing With Multi-Window and Stochastic Viewport Prediction," *IEEE Trans. Mob. Comput.*, vol. 24, no. 7, pp. 5903–5915, 2025, doi: 10.1109/TMC.2025.3541748.
- [17] A. Iskandar, M. Al-Sada, T. Miyake, Y. Saraiji, O. Halabi, and T. Nakajima, "Piton: Investigating the Controllability of a Wearable Telexistence Robot," *Sensors*, vol. 22, no. 21, Nov. 2022, doi: 10.3390/s22218574.
- [18] X. Corbillon, F. De Simone, and G. Simon, "360-degree video head movement dataset," in *Proceedings of the 8th ACM Multimedia Systems Conference, MMSys 2017*, Association for Computing Machinery, Inc, Jun. 2017, pp. 199–204. doi: 10.1145/3083187.3083215.
- [19] W. C. Lo, C. L. Fan, J. Lee, C. Y. Huang, K. T. Chen, and C. H. Hsu, "360° video viewing dataset in head-mounted virtual reality," in *Proceedings of the 8th ACM Multimedia Systems Conference, MMSys 2017*, Association for Computing Machinery, Inc, Jun. 2017, pp. 211–216. doi: 10.1145/3083187.3083219.

- [20] Y. Bao, H. Wu, T. Zhang, A. A. Ramli, and X. Liu, "Shooting a moving target: Motion-prediction-based transmission for 360-degree videos," in *Proceedings - 2016 IEEE International Conference on Big Data, Big Data 2016*, Institute of Electrical and Electronics Engineers Inc., 2016, pp. 1161–1170. doi: 10.1109/BigData.2016.7840720.
- [21] C. Wu, Z. Tan, Z. Wang, and S. Yang, "A dataset for exploring user behaviors in VR spherical video streaming," in *Proceedings of the 8th ACM Multimedia Systems Conference, MMSys 2017*, Association for Computing Machinery, Inc, Jun. 2017, pp. 193–198. doi: 10.1145/3083187.3083210.
- [22] Y. Guan, C. Zheng, X. Zhang, Z. Guo, and J. Jiang, "Pano: Optimizing 360° video streaming with a better understanding of quality perception," in *SIGCOMM 2019 - Proceedings of the 2019 Conference of the ACM Special Interest Group on Data Communication*, Association for Computing Machinery, Aug. 2019, pp. 394–407. doi: 10.1145/3341302.3342063.
- [23] A. T. Nasrabadi *et al.*, "A taxonomy and dataset for 360° videos," in *Proceedings of the 10th ACM Multimedia Systems Conference, MMSys 2019*, Association for Computing Machinery, Jun. 2019, pp. 273–278. doi: 10.1145/3304109.3325812.
- [24] K. Cho *et al.*, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, Association for Computational Linguistics (ACL), 2014, pp. 1724–1734. doi: 10.3115/v1/d14-1179.
- [25] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *3rd International Conference on Learning Representations (ICLR)*, 2015. [Online]. Available: <https://arxiv.org/abs/1412.6980>
- [26] A. E. Hoerl and R. W. Kennard, "Ridge Regression: Biased Estimation for Nonorthogonal Problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970, doi: 10.1080/00401706.1970.10488634.
- [27] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
- [28] A. Vaswani *et al.*, "Attention Is All You Need," in *Advances in Neural Information Processing Systems*, vol. 30, pp. 5998–6008, 2017. [Online]. Available: <https://arxiv.org/abs/1706.03762>.