

PERMAINAN VIRTUAL BALAP KELERENG MENGGUNAKAN ALGORITMA KALMAN FILTER

TRADITIONAL MARBLES RACING GAME BASED ON VIRTUAL REALITY USING KALMAN FILTER ALGORITHM

Kadek Dwi Suryana Putra¹, Tito Waluyo Purboyo², Anton Siswo Raharjo Ansori³

^{1,2,3} Prodi S1 Teknik Komputer, Fakultas Teknik Elektro, Universitas Telkom
¹dwisuryana@student.telkomuniversity.ac.id, ² titowaluyo@telkomuniversity.ac.id,
³ raharjo@telkomuniversity.ac.id

Abstrak

Perkembangan teknologi yang semakin maju menyebabkan beberapa bidang industri semakin berinovasi membuat sesuatu yang dapat menarik perhatian, salah satunya adalah *game* berbasis *virtual Reality*. *Virtual reality* merupakan teknologi yang membuat pengguna untuk dapat berinteraksi dengan dunia maya yang disimulasikan oleh komputer dengan menggabungkan objek 3D di dalamnya. Permainan kelereng termasuk salah satu permainan tradisional yang sekarang sudah semakin hilang di kalangan anak-anak yang lebih memilih untuk bermain dengan menggunakan *smartphone*. Dalam penelitian tugas akhir ini dirancang sebuah permainan tradisional balap kelereng berbasis *virtual reality* dengan menggunakan algoritma Kalman Filter yang berguna untuk mengetahui getaran dan posisi saat kelereng jatuh dengan menggunakan sensor mpu6050. *Unity3D* digunakan untuk membuat model simulasi lingkungan dengan menggunakan C# sebagai bahasa pemrograman yang diimplementasikan ke dalam *game* dengan menggunakan parameter dengan level tertentu, yaitu level 1, level 2, dan level 3. Pada penelitian ini menggunakan dua pengujian pitch dan roll masing masing menghasilkan standar deviasi pada sudut 60 derajat. Pengujian pitch dengan menggunakan filter yaitu 0,010954 sedangkan tanpa filter 0,140327 sedangkan pengujian roll menggunakan filter yaitu 0,008367, sedangkan tanpa filter 0,663378.

Kata kunci: *Kalman Filter, virtual reality, Unity3D*

Abstract

The development of increasingly advanced technology causes some industry fields to innovate to create something that can attract attention, one of which is virtual reality-based games. Virtual reality is a technology that allows users to interact with virtual worlds simulated by computers by combining 3D objects in it. Marbles games are one of the traditional games that are now increasingly lost among children who prefer to play using smartphones. In this final task researcher has designed a traditional game of marble racing based on virtual reality using Kalman Filter algorithm that is useful to know the vibration and position when marbles fall using sensor mpu6050. Unity3D is used to create an environmental simulation model using C# as the programming language that is implemented into the game using parameters with a certain level, namely level 1, level 2, level 3. In this research, contains two tests pitch and roll each produced a standard deviation at an angle of 60 degrees. The pitch test using a filter is 0.010954 while without a filter is 0.140327 and the roll test uses a filter that is 0.008367, while without a filter is 0.663378.

Keywords: *Kalman Filter, virtual reality, Unity3D*

1. Pendahuluan

Pada zaman teknologi yang semakin canggih permainan tradisional semakin ditinggalkan digantikan oleh permainan pada telepon genggam yang lebih canggih padahal permainan tradisional sangat asyik dimainkan selain melatih sistem motorik anak, permainan tradisional juga mudah untuk melatih interaksi sosial dengan lingkungan secara langsung. Dengan perkembangan teknologi yang sekarang sedikit anak – anak ingin memainkan permainan tradisional khususnya di perkotaan. Penyebabnya adalah akulturasi budaya modern, banyak permainan digital, sangat sedikit tempat bermain[1]. Permainan tradisional yang masih sering dimainkan adalah balap kelereng. Balap kelereng adalah permainan yang biasa dimainkan untuk merayakan hari kemerdekaan Indonesia. Cara memainkan permainan ini sangat sederhana dimana permainan ini membutuhkan 1 buah sendok dan 1 butir kelereng. Masing-masing harus berusaha menjaga keseimbangan agar kelereng yang dibawa tidak terjatuh ke tanah, siapa yang pertama mencapai garis *finish* dan berhasil menjaga kelereng agar tidak terjatuh ke tanah maka dia akan menjadi pemenangnya.

Salah satu metode yang digunakan untuk membuat permainan ini dapat dimainkan lagi yaitu dengan membawa permainan kelereng ini ke dalam dunia digital atau virtual. Metode tersebut dikenal dengan nama “*Virtual Reality*”

atau disingkat VR teknologi ini dapat membawa pengguna masuk ke dalam dunia *virtual* yang disimulasikan oleh komputer untuk memanipulasi otak manusia sehingga pengguna merasakan berbagai hal yang terjadi pada dunia virtual seolah-olah seperti dunia nyata[1].

Pada penelitian ini membuat simulasi balap kelereng berbasis virtual reality dengan menggunakan metode Kalman Filter. Di mana, metode *Kalman* Filter digunakan untuk mengurangi *noise* pada sensor untuk menggerakkan objek 3D berupa sendok pada *game*.

2. Dasar Teori.

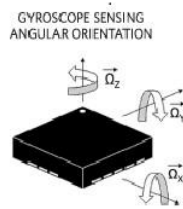
2.1 Permainan Balap Kelereng

Dalam perayaan hari kemerdekaan Indonesia di berbagai daerah pasti selalu mengadakan beragam perlombaan yang dapat diikuti oleh segala usia. perlombaan yang diadakan ada banyak jenisnya salah satunya lomba balap kelereng, permainan ini selalu dimainkan pada 17 agustus untuk memeriahkan hari kemerdekaan Indonesia karena itu permainan lomba balap kelereng ini dapat dibilang permainan yang memiliki umur yang sudah cukup tua dan permainannya dikenal oleh seluruh masyarakat Indonesia.

Permainan balap kelereng adalah permainan yang sifatnya lomba lari dengan membawa kelereng yang diletakan di atas sendok sambil menggigit pangkal dari sendok, permainan balap kelereng biasanya di mainkan lebih dari dua orang atau sekurang - kurangnya dapat dimainkan oleh dua orang yang berjalan cepat menuju garis *finish* dengan menjaga agar kelereng yang di letakan pada sendok tidak terjatuh, pemain yang terlebih dahulu mencapai garis *finish* maka dinyatakan sebagai pemenangnya namun jika kelereng yang diletakan pada sendok terjatuh sebelum mencapai garis *finish* maka yang bersangkutan dinyatakan gugur/kalah.

2.2 Gyroscope

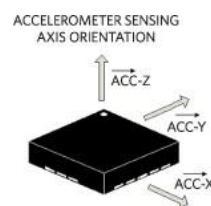
Pada *gyroscope* perangkat sensor yang dinamakan *gyro* sensor yang digunakan untuk menentukan orientasi gerak dengan bertumpu pada roda atau cakram yang berotasi atau berputar pada satu sumbu tertentu. Sensor ini berfungsi untuk mengidentifikasi Gerakan sesuai dengan gravitasi. Sebelum digunakan sensor *gyro* harus melakukan kalibrasi untuk menentukan acuan orientasi yang digunakan sebagai dasar. Sensor pada *gyroscope* memiliki keluaran berupa kecepatan sudut dari 3 arah sumbu, yaitu sumbu x, sumbu y, dan sumbu z. sumbu x akan menjadi sudut phi yang mengidentifikasi arah kanan dan kiri. Sumbu y, disisi lain, akan menjadi sudut theta yang mengidentifikasi aras atas dan bawah. Sumbu z akan menjadi sudut psi yang mengidentifikasi arah depan dan belakang [6].



Gambar 2. 1 Gyroscope

2.3 Sensor Accelerometer

Accelerometer adalah sensor yang digunakan untuk mengukur percepatan suatu benda. Fungsi *accelerometer* adalah untuk mengukur percepatan dinamis dan statis. Pengukuran dinamis adalah pengukuran percepatan suatu benda yang bergerak, sedangkan pengukuran statis adalah pengukuran gravitasi bumi untuk mengukur sudut kemiringan. Pada *accelerometer* sensor dapat di implementasikan pada peralatan elektronik seperti, *mouse* komputer dan telepon. Pada *accelerometer* percepatan adalah suatu kondisi dimana terjadi perubahan kecepatan terhadap waktu, jika bertambahnya satu kecepatan dalam rentang waktu tertentu maka disebut percepatan (*acceleration*).



Gambar 2. 2 Sensor Accelerometer

2.4 Kalman Filter

Kalman *Filter* adalah sebuah algoritma yang dapat mengontrol sistem yang *sensitive* terhadap *noise* yang berguna untuk mengurangi *error* dan dapat mengestimasi kondisi tertentu. Pada filter ini perhitungan pengukuran yang terpapar *noise* dari sensor dapat dikurangi sebelum masuk ke dalam proses sistem *control*. Ada dua tahapan pada proses Kalman *Filter* yaitu, *predict* dan *update*. Untuk tahap *predict* yaitu menggunakan estimasi keadaan pada waktu sebelumnya agar dapat mengestimasi keadaan sekarang. Selanjutnya, pada tahap kedua yaitu, *update* yang digunakan untuk memperbaiki tahap pertama agar mendapatkan estimasi kemungkinan yang lebih akurat. Berikut persamaan pada Kalman Filter:

1. Predict

$$\hat{x}_{t|t-1} = F_t \hat{x}_{t-1|t-1} + B_t \tau_t \quad (2.1)$$

$$P_{t|t-1} = F_t P_{t-1|t-1} F_t^T + Q_t \quad (2.2)$$

2. Update

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t (y_t - H_t \hat{x}_{t|t-1}) \quad (2.3)$$

$$K_t = P_{t|t-1} H_t^T (H_t P_{t|t-1} H_t^T + R_t)^{-1} \quad (2.4)$$

$$P_{t|t} = (I - K_t H_t) P_{t|t-1} \quad (2.5)$$

2.5 ESP32s

ESP32s adalah *mikrocontroller* yang terintegrasi dengan 2.4 GHz *Wi-Fi*, *Bluetooth* dan 4 MB *flash memory* yang memiliki 38 GPIO pin. Dalam modul ini, selain ESP32s *mikrocontroller* ini memiliki kristal 40MHz selain kristal 160MHz dan 240MHz juga memiliki antena internal dan *port micro USB* untuk daya dan pemrograman, ditambah dengan 36 pin *General Purpose Input/Output (GPIOs)*.



Gambar 2. 3 Mikrocontroller ESP32s

2.6 MPU-6050

MPU-6050 adalah sensor yang memiliki 9 axis *motion sensor* dengan 3 axis *gyroscope* dan 3 axis *accelerometer* dan digital motion processor DMP, ILC atau SPI interface yang dapat dikoneksikan dengan sensor lain dengan total 9 *output* sinyal axis. Pada sensor MPU6050 dapat mendeteksi objek dengan sudut *horizontal* maupun sudut *vertical* atau dapat membaca indikasi kemiringan sudut objek dari data pada sensor *accelerometer* dan *gyroscope*.



Gambar 2. 4 MPU-6050

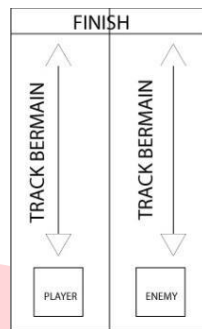
2.7 Arduino IDE

Arduino IDE adalah *software* yang berguna untuk meng-*compile* serta mengunggah program ke *board* Arduino dengan bahasa java, pada *software* Arduino IDE memiliki beberapa menu yaitu editor program, adalah tempat pengguna menulis kode ataupun mengedit kode yang nantinya akan di *compile* ke dalam *board*, menu *compile* berguna untuk mengubah kode yang sudah ditulis untuk dijadikan kode biner yang dapat dibaca oleh *microcontroller*, *uploader* berfungsi untuk memasukkan kode biner dari komputer ke dalam *memory* pada *board* Arduino.

3. Perancangan Sistem

3.1 Gambaran Umum Sistem

Algoritma Kalman Filter ini digunakan untuk menghitung kumpulan persamaan dan data yang telah didapat untuk mengestimasi hasil sebenarnya dari posisi dan kecepatan yang terjadi pada sendok. *Study case* yang dibahas dalam tugas akhir ini adalah membuat permainan balap kelereng di dalam sebuah ruangan 3D yang akan di implementasikan ke dalam *Virtual Reality* dengan memiliki *level* dalam berupa jarak atau tingkat kesulitan di dalam permainan balap kelereng tersebut. Berikut gambaran umum sistem pada tugas akhir ini adalah sebagai berikut:

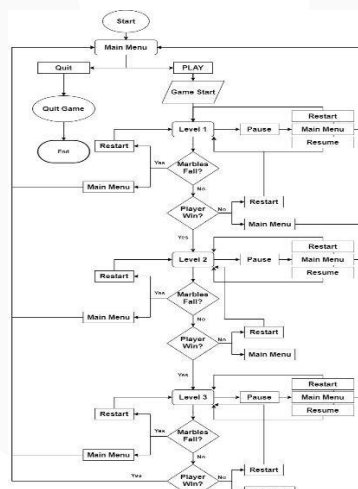


Gambar 3. 1 Gambaran Umum Sistem

Saat *game* dimulai, *player* akan diberikan waktu 3 detik untuk melakukan persiapan lalu bergerak sambil menjaga keseimbangan agar kelereng yang berada pada sendok tidak terjatuh ke tanah sambil menuju garis *finish* yang telah ditentukan pada setiap level, jika *player* sudah berhasil mencapai garis *finish* maka *player* dapat melanjutkan ke level selanjutnya, namun jika *player* kalah terhadap *bot* yang telah disiapkan maka *player* dapat melakukan pertandingan ulang

3.2 Flowchart Firebase

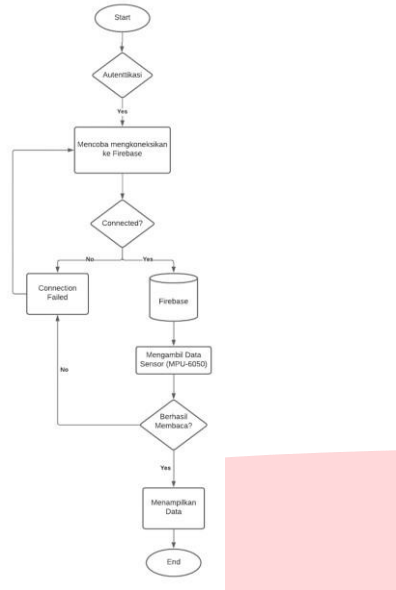
3.3.1 Flowchart Gameplay



Gambar 3. 2 Flowchart Gameplay

Tampilan yang akan dilihat pertama kali oleh *user* pada *game* ini adalah main menu yang terdapat dua tombol yaitu *play* dan *quit*. Tombol *quit* akan mengarahkan *user* untuk keluar dari permainan yang berarti prosesnya berakhir, lalu untuk tombol *play* akan masuk ke proses selanjutnya yaitu level 1. Dalam setiap level permainan terdapat kondisi saat *user* melakukan pause untuk menghentikan permainan sementara, lalu dalam proses ini akan terdapat tiga tombol yang terdiri dari resume untuk memulai Kembali proses yang sedang dihentikan, main menu untuk Kembali ke tampilan utama, dan *restart* untuk mengulang level yang sedang dimainkan. Dalam setiap level juga memiliki dua kondisi untuk *user* dinyatakan kalah yaitu saat kelereng terjatuh ke tanah atau lawan bermain sampai ke garis *finish* terlebih dahulu, jika kelereng jatuh ke tanah atau lawan bermain sampai ke garis *finish* terlebih dahulu maka *user* dapat memilih untuk mengulang Kembali permainan pada level tersebut atau Kembali ke main menu. Selanjutnya apabila *user* dinyatakan menang maka *user* diarahkan ke *level* selanjutnya atau Kembali ke *main menu*. Pada permainan ini hanya terdapat tiga level jika sudah menyelesaikan level tiga maka *user* Kembali ke *main menu*.

3.3.2 Flowchart Firebase

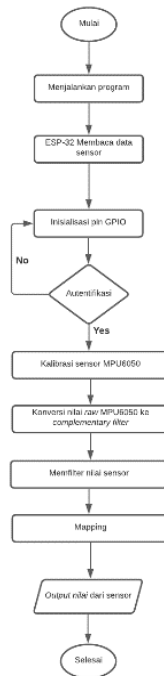


Gambar 3. 3 Flowchart Firebase

Pada gambar di atas menjelaskan tentang alur dari pengiriman data sensor untuk dapat di tampung pada *database* yang ada di dalam Firebase untuk dapat mengimplementasikan *output* dari sensor yang datanya akan di baca pada Unity.

3.4 Perancangan Alur Kalman Filter

Perancangan Kalman *filter* dengan menggunakan sensor MPU6050 yang digunakan sebagai penyimpan data yang nantinya dilakukan proses *filtering* yang dilakukan pada platform Arduino *IDE* untuk menampilkan sinyal plot sebelum dan sesudah dilakukannya *filtering*.

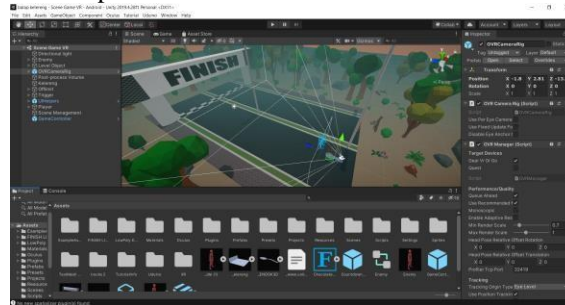


Gambar 3. 4 Flowchart Kalman Filter

4. Pengujian dan Analisis

4.1 Implementasi Game

Implementasi ini sebagai proses penerapan dari rancangan yang telah dibuat untuk dapat menjadi suatu aplikasi yang dapat dijalankan dan dioperasikan.



Gambar 4. 1 Tampilan Awal Unity

Pada gambar di atas terdapat beberapa jendela seperti *hierarchy* yang berfungsi sebagai inisiasi objek yang berada pada *scene* di dalam permainan, *assets* yang berfungsi sebagai penyimpanan dari objek-objek maupun kodingan yang telah dibuat, *inspector* berfungsi sebagai jendela untuk mengubah nilai dari *property* dan *setting*, *inspector* juga dapat menampilkan informasi dari *player setting*, *assets*, *game object*, *scene*, *prefab*, dan lain-lain.

4.2 Implementasi Pengujian Kalman Filter Pada sensor MPU-6050

4.2.1 Pengujian Pitch

Pengujian *Pitch* (Y) bertujuan menguji hasil *output* dari sensor untuk mengetahui perbedaan sebelum dan sesudah di lakukan *filtering* menggunakan *Kalman Filter*. Nilai yang di gunakan pada pengujian berdasarkan *output* serial monitor pada Arduino IDE dengan sudut 0°, 30°, 60°, 90°. Pengujian ini dilakukan dengan membandingkan Standar Deviasi (SD) sensor non *filter* dan filter dengan melakukan dua kali pengujian.

Tabel 4. 1 Pengujian Pertama *Pitch* dengan 5 Data

Pengujian Roll				
Tanpa Filter				
No	0°	30°	60°	90°
Sampel Ke- 1	0,62	30,81	60,44	90,43
Sampel Ke- 2	0,29	30,57	60,57	90,21
Sampel Ke- 3	0,63	30,51	60,56	90,32
Sampel Ke- 4	0,29	30,68	60,73	90,07
Sampel Ke- 5	0,43	30,72	60,43	90,13
Rata - Rata	0,452	30,658	60,5425	90,232
SD	0,167988	0,119457	0,140327	0,144983
Kalman Filter				
No	0°	30°	60°	90°
Sampel Ke- 1	0,66	30,99	60,59	90,29
Sampel Ke- 2	0,66	30,98	60,59	90,29
Sampel Ke- 3	0,66	30,98	60,61	90,29
Sampel Ke- 4	0,65	30,97	60,61	90,3
Sampel Ke- 5	0,65	30,97	60,61	90,29
Rata - Rata	0,656	30,978	60,602	90,292
SD	0,005477	0,008367	0,010954	0,004472

Tabel 4. 2 Pengujian Kedua *Pitch* dengan 5 Data

Pengujian Roll				
Tanpa Filter				
No	0°	30°	60°	90°
Sampel Ke- 1	0,22	30,57	60,85	90,09
Sampel Ke- 2	0,23	30,71	60,76	90,29
Sampel Ke- 3	0,63	30,81	60,69	90,12
Sampel Ke- 4	1,05	31,28	60,03	90,06
Sampel Ke- 5	0,58	30,99	60,31	90,01
Rata - Rata	0,542	30,872	60,528	90,114
SD	0,342155	0,274627	0,34615	0,106442
Kalman Filter				
No	0°	30°	60°	90°
Sampel Ke- 1	0,63	30,96	60,61	90,28
Sampel Ke- 2	0,62	30,95	60,61	90,28
Sampel Ke- 3	0,63	30,95	60,61	90,27
Sampel Ke- 4	0,63	30,96	60,6	90,28
Sampel Ke- 5	0,63	30,96	60,61	90,28
Rata - Rata	0,628	30,956	60,608	90,278
SD	0,004472	0,005477	0,004472	0,004472

Dari dua pengujian *pitch* yang dilakukan bahwa sensor dengan *filter* lebih tahan terhadap noise dibandingkan dengan tidak menggunakan *filter*. Hal tersebut ditunjukkan pada nilai standar deviasi atau data akuisisi sensor dengan *filter* lebih kecil dibandingkan sensor tanpa filter. Pada sudut 60 derajat didapatkan hasil variasi *noise* sebesar 0,119457 derajat pada sensor dengan *filter* sebesar 0,010954 derajat

4.2.2 Pengujian Roll

Pengujian *Roll* (X) bertujuan untuk menguji hasil *output* dari sensor untuk mengetahui perbedaan sebelum dan sesudah di lakukan *filtering* menggunakan *Kalman Filter*. Nilai yang di gunakan pada pengujian berdasarkan *output* serial monitor pada Arduino IDE dengan sudut 0°, 30°, 60°, 90°. Perbedaan dengan pengujian *pitch* adalah pengujian *roll* bergerak sesuai dengan sumbu X, sedangkan pengujian *pitch* bergerak sesuai sumbu Y. Pengujian ini dilakukan dengan membandingkan Standar Deviasi (SD) sensor non filter dan filter dengan melakukan 2 kali pengujian.

Tabel 4. 3 Pengujian Pertama *Roll* dengan 5 Data

Pengujian Roll				
Tanpa Filter				
No	0°	30°	60°	90°
Sampel Ke- 1	0,14	30,35	60,95	90,53
Sampel Ke- 2	0,25	30,21	59,36	90,22
Sampel Ke- 3	0,17	30,31	60,36	90,5
Sampel Ke- 4	0,29	30,69	60,23	90,79
Sampel Ke- 5	0,64	30,29	60,99	90,62
Rata - Rata	0,2125	30,37	60,378	90,532
SD	0,069462	0,186011	0,663378	0,207774
Kalman Filter				
No	0°	30°	60°	90°
Sampel Ke- 1	0,25	30,41	60,53	90,57
Sampel Ke- 2	0,25	30,42	60,52	90,56
Sampel Ke- 3	0,25	30,41	60,51	90,56
Sampel Ke- 4	0,26	30,41	60,51	90,56
Sampel Ke- 5	0,25	30,42	60,52	90,57
Rata - Rata	0,252	30,414	60,518	90,564
SD	0,004472	0,005477	0,008367	0,005477

Tabel 4. 4 Pengujian Kedua *Roll* dengan 5 Data

Pengujian Roll				
Tanpa Filter				
No	0°	30°	60°	90°
Sampel Ke- 1	0,29	30,27	60,95	90,59
Sampel Ke- 2	0,14	30,48	60,36	90,03
Sampel Ke- 3	0,36	30,76	60,23	90,34
Sampel Ke- 4	0,17	30,58	60,21	90,18
Sampel Ke- 5	0,26	30,23	60,36	90,69
Rata - Rata	0,244	30,464	60,422	90,366
SD	0,08961	0,220068	0,30343	0,275372
Kalman Filter				
No	0°	30°	60°	90°
Sampel Ke- 1	0,25	30,48	60,53	90,49
Sampel Ke- 2	0,24	30,48	60,52	90,48
Sampel Ke- 3	0,24	30,49	60,51	90,48
Sampel Ke- 4	0,24	30,49	60,51	90,47
Sampel Ke- 5	0,25	30,49	60,52	90,47
Rata - Rata	0,244	30,486	60,518	90,478
SD	0,005477	0,005477	0,008367	0,008367

Dari pengujian roll yang dilakukan dengan membandingkan antara pengujian roll pertama dengan kedua menunjukkan hasil serupa bahwa sensor dengan filter lebih tahan terhadap *noise* dibandingkan dengan yang tidak menggunakan filter. hal tersebut ditunjukkan pada nilai hasil standar deviasi atau data akuisisi sensor dengan filter lebih kecil dibandingkan sensor tanpa filter, yakni dengan nilai pada sudut 60 derajat di dapatkan standar deviasi sensor menggunakan filter lebih kecil yaitu, 0,008367, sedangkan tanpa filter 0,663378.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Dari pengujian dan analisis yang sudah dilakukan oleh penulis terhadap sistem yang sudah dirancang dapat diambil kesimpulan sebagai berikut:

1. Implementasi algoritma Kalman *Filter* Dapat diimplementasikan kepada Sensor MPU- 6050. Dilakukan 2 pengujian dengan menggunakan pitch dan roll. Masing - masing pengujian dilakukan 2 kali pengambilan data dengan jumlah data sebanyak 5 pada masing - masing pengujian.
2. Pada pengujian *pitch* menunjukkan bahwa akuisisi data sensor menggunakan filter lebih unggul dibandingkan sensor tanpa menggunakan filter yakni ditunjukkan dengan hasil *noise* sensor yang menggunakan Kalman filter lebih kecil dibandingkan sensor tanpa menggunakan filter .Standar deviasi sensor pakai filter pada 60 derajat yaitu 0,010954 sedangkan tanpa filter 0,140327.
3. Pada pengujian *roll* di dapatkan standar deviasi sensor menggunakan filter lebih kecil yaitu, 0,008367, sedangkan tanpa filter 0,663378.

5.2 Saran

Untuk penelitian lebih lanjut, penulis memiliki saran yang nantinya dapat diimplementasikan untuk penelitian selanjutnya

1. Penelitian selanjutnya diharapkan *database* sistem dapat dihubungkan menggunakan *local server* sehingga *delay* yang terjadi dapat di atasi.
2. Diharapkan untuk peneliti selanjutnya dapat meningkatkan akurasi Kalman filter untuk mengurangi *noise* yang terjadi pada sensor.

Referensi

- [1] A. Setiawan, A. S. Nugraha, H. Haryanto dan I. Gamayanto, "Game Virtual Reality Turn-Based Untuk Pelatihan Permainan Tradisional Benthik," dalam *Prosiding Seminar Nasional Geotik*, 2018..
- [2] M. Uswatun Hasnah, "Pengembangan Kemampuan Fisik Motorik Melalui Permainan Tradisional Bagi Anak Usia Dini," *Jurnal Pendidikan Anak*, 2016.
- [3] D. Zillmann dan P. Vorderer, *Media Entertainment The Psychology Of Its Appeal*, New York: Lawrence Erlbaum Associates, Inc, 2009.
- [4] K. Squire, "Video Games and Learning: Teaching and Participatory Culture in the Digital Age," dalam *Video Games and Learning: Teaching and Participatory Culture in the Digital Age*, Gainesville, Teacher College Press, 2011, p. 101.
- [5] R. Rouse, *Game Design: Theory & Practice Second Edition*, Sudbury: Wordware Publishing, Inc., 2005.
- [6] B. Kominfo, "GYROSCOPE SENSOR SEBAGAI 'UPGRADE' DARI ACCELEROMETER," Agustus 2019.
- [7] H. Riyadi, "nesabamedia," [Online]. Available: <https://www.nesabamedia.com/pengertian-vr-virtual-reality/>. [Diakses 28 November 2020].
- [8] J. Petty, "conceptartempire," [Online]. Available: <https://conceptartempire.com/what-is-unity/>. [Diakses 28 November 2020].
- [9] S. Nakov, *Fundamentals of Computer Programming with C#*, Bulgaria: Svetlin Nakov & Co, 2013
- [10] D. K. Lee, J. In dan S. Lee, "Standard deviation and standard," *Korean Journal of Anesthesiology*, 2005

