

PENINGKATAN KECERDASAN COMPUTER PLAYER PADA GAME PERTARUNGAN BERBASIS K-NEAREST NEIGHBOR BERBOBOT

M. Ihsan Alfani Putera¹⁾ dan Darlis Herumurti²⁾

^{1, 2)}Departemen Informatika, Fakultas Teknologi Informasi dan Komunikasi, Institut Teknologi Sepuluh Nopember
Kampus ITS, Jl. Raya ITS, Sukolilo – Surabaya 60111, Telp. +62 31 5939214, Fax. +62 31 5913804
e-mail: m.ihsan.ap@gmail.com¹⁾, darlis@if.its.ac.id²⁾

ABSTRAK

Salah satu teknologi komputer yang berkembang dan berubahannya cukup pesat adalah game. Tujuan dibuatnya game adalah sebagai sarana hiburan dan memberikan kesenangan bagi penggunanya. Contoh elemen dalam pembuatan game yang penting adalah adanya tantangan yang seimbang sesuai level. Dalam hal ini, adanya kecerdasan buatan atau AI merupakan salah satu unsur yang diperlukan dalam pembentukan game. Penggunaan AI yang tidak beradaptasi ke strategi lawan akan mudah diprediksi dan repetitif. Jika AI terlalu pintar maka player akan kesulitan dalam memainkan game tersebut. Dengan keadaan seperti itu akan menurunkan tingkat enjoyment dari pemain. Oleh karena itu, dibutuhkan suatu metode AI yang dapat beradaptasi dengan kemampuan dari player yang bermain. Sehingga tingkat kesulitan yang dihadapi dapat mengikuti kemampuan pemainnya dan pengalaman enjoyment ketika bermain game terus terjaga. Pada penelitian sebelumnya, metode AI yang sering digunakan pada game berjenis pertarungan adalah K-NN. Namun metode tersebut menganggap semua atribut dalam game adalah sama sehingga hal ini mempengaruhi hasil learning AI menjadi kurang optimal.

Penelitian ini mengusulkan metode untuk AI dengan menggunakan metode K-NN berbobot pada game berjenis pertarungan. Dimana, pembobotan tersebut dilakukan untuk memberikan pengaruh setiap atribut dengan bobot disesuaikan dengan aksi player. Dari hasil evaluasi yang dilakukan terhadap 50 kali pertandingan pada 3 skenario uji coba, metode yang diusulkan yaitu K-NN berbobot mampu menghasilkan tingkat kecerdasan AI dengan akurasi sebesar 51%. Sedangkan, metode sebelumnya yaitu K-NN tanpa bobot hanya menghasilkan tingkat kecerdasan AI sebesar 38% dan metode random menghasilkan tingkat kecerdasan AI sebesar 25%.

Kata Kunci: Computer player, game, game pertarungan, kecerdasan buatan, K-NN berbobot

ABSTRACT

One of the computer technology that develops and changes quite rapidly is game. The purpose of game creation is as entertainment facility which gives pleasure to its users. The example of the important element in game creation is balance challenge by level. In this case, the existence of artificial intelligence or AI is one of the elements which is needed in game formation. The unadaptive AI use will be easily predicted by the opponent and repetitive. If AI is too smart, the player will have difficulty in playing the game. Consequently, it will reduce the level of the players' enjoyment. Therefore, it needs an adaptive AI method that can adapt the capabilities of the players. So that the difficulty level can be arranged automatically by following the player's ability and enjoyment experience during the continuous play. Previous studies, AI method had been frequently used in game fighting type is K-NN. However, the method considers that all the attributes in the game are similar so it affects the learning of AI which can be less optimal.

This study proposed a method for AI using the weighted K-NN method on game fighting type. In this study, the weighting was done by giving effect to each attribute with weight based on the player action. Based on the evaluation results of 50 times competition on 3 trial scenario, the proposed method, weighted K-NN was capable to result AI intelligence levels with the accuracy level about 51%. Meanwhile, the previous method of K-NN only resulted AI intelligence levels about 38%, while the random method resulted AI intelligence levels of AI about 25%.

Keywords: Artificial intelligence, computer player, fighting game, game, weighted K-NN.

I. PENDAHULUAN

SALAH satu teknologi yang tingkat perkembangannya sangat pesat yaitu komputer. Hampir semua aspek kehidupan manusia sangat bergantung dengan teknologi komputer. Dengan kemajuan teknologi, hampir semua konten, proses kerja dan interaksi fisik dapat dilakukan melalui proses digital. Salah satu produk dari teknologi komputer yang berkembang dan berubahannya cukup pesat adalah game. Secara umum tujuan dibuatnya game adalah sebagai sarana hiburan dan untuk memberikan kesenangan bagi penggunanya. Hal tersebut merupakan salah satu faktor yang mendorong tingkat perkembangan industri game. Dalam pembuatan game, pasti memerlukan suatu perancangan yang sesuai dengan tujuan pembuat. Salah satu contoh elemen yang penting ialah adanya tantangan yang seimbang sesuai level [9]. Dalam hal ini, adanya kecerdasan buatan atau AI (*Artificial intelligence*) merupakan salah satu unsur yang diperlukan dalam pembentukan game. Melihat potensi dari *digital game* yang cukup tinggi, banyak game yang dibuat sebagai perlombaan untuk kalangan pecinta game. Salah satu jeni game yang sering digunakan dalam pertandingan adalah game berjenis pertarungan atau *fighting game* [9].

Game pertarungan adalah bentuk game yang memiliki unsur hiburan dan tantangan. Game tersebut merupakan

game bergenre *Turn-Based Role Playing Game* yang merupakan sub *genre* dari *Role Playing Game* (RPG). Pada *turn based* RPG, pemain mengontrol satu atau lebih karakter dengan berbagai aksi yang dimiliki. Pertempuran dalam *genre* tersebut dilakukan dengan cara penentuan giliran. Ada banyak variasi dari cara penentuan giliran. Salah satu contohnya adalah permainan catur yaitu penentuan giliran selanjutnya dilakukan setelah pemain sebelumnya telah menyelesaikan aksinya. Dalam tiap giliran, pemain memilih aksi yang dilakukan oleh tiap karakter dalam permainan yang sedang dikontrol. Aksi ini bisa berupa mengeluarkan aksi menyerang, bergerak, menggunakan item, dan aksi lainnya. Misi dari tiap pertempuran bermacam-macam misalnya mengeliminasi musuh, bertahan, dll. Tapi pada umumnya misi tiap pertempuran adalah mengeliminasi musuh yang dihadapi [12]. Hal serupa juga dilakukan pada *game* pertarungan, misi utama dalam permainan ini adalah mengeliminasi musuh dengan aksi yang dimiliki.

Pada dasarnya *game* AI merupakan *game* yang mengubah metode, proses, dan algoritma pada kecerdasan tersebut yang akan diaplikasikan ke pembuatan dan pengembangan *game*. Terdapat tiga panorama dalam *game* AI yakni perspektif metode (komputer), perspektif pengguna (manusia) dan perspektif interaksi pemain [11]. Hal ini menunjukkan bahwa kecerdasan buatan dan kecerdasan komputasional sering digunakan pada kebanyakan *game* dan saling memiliki ketergantungan interaksi terhadap pemain. Sehingga AI memiliki peran yang penting untuk meningkatkan ketertarikan pengguna dalam bermain *game*.

Saat ini masih terdapat pengembang *game* yang masih menggunakan AI berdasarkan *rule based system*. Metode yang digunakan biasanya akan menghasilkan *behaviour* yang statis dan tidak beradaptasi ke strategi lawan. AI dengan kondisi yang seperti itu akan terlalu mudah diprediksi dan repetitif. Jika pemain mampu menghafalkan perilaku dari AI *game*, pemain akan kehilangan minatnya setelah bermain beberapa *game* dikarenakan tantangan yang kurang menarik [9]. Hal seperti ini yang nantinya dapat menurunkan tingkat *enjoyment* pemain dalam bermain *game*. Berdasarkan hal tersebut banyak penelitian yang ditujukan untuk memanfaatkan teknik *learning* serta jenis teknik lainnya dalam menciptakan agen AI yang mampu beradaptasi dengan *player*.

Beberapa penelitian terkait *game* adaptif AI dilakukan [4], peneliti melakukan penelitian tersebut untuk menghasilkan kecerdasan buatan yang adaptif pada *game* Shogi [6]. Dari hasil penelitian didapatkan hasil bahwa kecerdasan buatan yang dibuat mampu menghasilkan *computer player* yang beradaptasi dengan kebiasaan *player*. Penelitian lain juga dilakukan [5], dimana peneliti menggunakan metode berbasis adaptif pada *game* Capture The Flag. *Game* tersebut merupakan salah satu *game* yang kompleks dan *adversarial*, dengan pemain yang saling berkompetisi dan tujuan yang membutuhkan keputusan untuk dibuat di level permainan yang berbeda. Dari penelitian terkait *game* adaptif AI tujuan utama dari penerapannya dalam berbagai jenis *game* adalah untuk memberikan kondisi yang *enjoyment* terhadap pemain. *Enjoyment* dapat terjadi apabila tercipta keseimbangan antara tantangan dalam *game* dan kemampuan yang dimiliki pemain.

Dari uraian di atas menunjukkan bahwa saat ini dibutuhkan suatu AI yang mampu beradaptasi dengan kemampuan *player*, dengan hal ini *player* akan mendapatkan sisi *enjoyment* dari permainan tersebut. AI dengan tingkat kecerdasan yang terlalu tinggi akan sulit dikalahkan oleh *player*, sehingga *player* akan mudah bosan. Metode AI yang sering digunakan dalam menciptakan AI yang dinamis adalah K-NN. Penelitian yang dilakukan [4] menggunakan metode K-NN pada *game* berjenis *action fighting*. Peneliti menggunakan metode tersebut untuk memprediksi gerakan *player* secara efektif agar pada *game fighting* tersebut tidak membosankan dan lebih menantang untuk dimainkan. Hasil percobaan yang telah dilakukan menunjukkan bahwa metode yang diusulkan belum mencapai hasil yang optimal, karena dalam penerapannya K-NN menganggap bahwa semua atribut dalam *game* tersebut berbobot sama. Dengan adanya faktor tersebut mempengaruhi akurasi AI yang dihasilkan.

Pada penelitian ini diusulkan metode untuk meningkatkan kecerdasan AI terhadap perilaku *player* dengan menerapkan metode K-NN berbobot pada *game* pertarungan berskala kecil. Dengan melakukan penelitian ini diharapkan dapat menjadi sebuah inovasi dalam bidang kecerdasan buatan atau AI, sehingga dapat tercapai tujuan yang diinginkan yaitu perilaku AI atau *computer player* mampu beradaptasi dengan kemampuan dari *player*.

II. KAJIAN PUSTAKA

A. *Game*

Game memiliki definisi yang bermacam-macam, dalam penjabaran definisi tersebut memiliki berbagai kesamaan dan perbedaan. *Game* adalah sebuah sistem di mana pemain terlibat dalam pertempuran buatan, ditentukan oleh aturan, yang menghasilkan hasil yang terukur [5]. Sedangkan menurut [3] *Game* dapat didefinisikan sebagai suatu bentuk permainan yang memiliki tujuan dan struktur. *Game* menyediakan aktivitas yang menarik minat pemain sekaligus sebagai media hiburan [2]. Berdasarkan pernyataan-pernyataan tersebut, aspek-aspek umum dan definisi *game* dapat dengan mudah diidentifikasi, yaitu *game* adalah permainan atau kegiatan yang memiliki seperangkat sistem aturan dan tujuan. Aspek lain yang menonjol di dalam *game* adalah adanya konflik atau pertempuran antara pemain dengan pemain lain maupun antara pemain dan sistem *game* itu sendiri.

B. Game Pertarungan

Game pertarungan merupakan *game* yang memiliki interaksi antara pemain dengan pemain lain maupun dengan sistem permainan tersebut yang dibatasi oleh aturan tertentu. Dalam hal ini *game* pertarungan dapat dikategorikan sebagai *role playing game*. *Game* diharuskan mengenali karakter, memiliki penekanan pada tokoh/peran perwakilan pemain di dalam permainan yang biasanya adalah tokoh utamanya, dimana seiring kita memainkannya, karakter tersebut dapat berubah dan berkembang ke arah yang diinginkan pemain (biasanya menjadi semakin hebat, semakin kuat, semakin berpengaruh, dll). Dalam hal ini *game* yang dilakukan memungkinkan pemain untuk dapat menentukan perencanaan strategi dan alur di dalam *game* untuk mencapai suatu tujuan tertentu [13]. Karakteristik dari *game* bergenis RPG yaitu kemungkinan untuk dapat mengembangkan peran karakter yang dipilih yang perkembangan dan keterampilannya melibatkan pengguna secara langsung di dalam permainan. Fitur inilah yang membuat *game* RPG menjadi lebih dinamis. Adanya kemajuan teknologi saat ini memungkinkan penggunaan komputer untuk memvisualisasikan permainan RPG sehingga memberikan pengalaman bermain dan realisme lebih kepada pemain. *Game* pertarungan menjadi salah satu *game* yang sering dimainkan bahkan untuk dipertandingkan [11]. *Player* akan merasakan sisi *enjoyment* dengan adanya *computer player* yang menerapkan berbagai macam metode AI, sehingga dengan adanya metode AI tersebut dapat meningkatkan tingkat *enjoyment* dari *player* sekaligus dapat beradaptasi dengan perilaku *player* yang memainkan *game* tersebut.

C. Artificial Intelligence (AI)

Kecerdasan buatan atau *Artificial Intelligence* (AI) merupakan ilmu tentang bagaimana membangun suatu sistem komputer yang menunjukkan kecerdasan dalam berbagai cara. AI merupakan area penelitian yang dinamis dalam topik riset ilmu komputer. Sampai saat ini, telah banyak penelitian mengenai perkembangan AI diantaranya *decision tree*, *neural network*, *evolutionary computing*, *machine learning*, *natural language processing*, dan *object oriented programming*. Banyak perkembangan terkait AI yang dibangun dalam permainan, salah satunya adalah AAI. *Adaptive Artificial Intelligence* (AAI) mengacu ke *non-player character* (NPC) dinamis dimana komputer mampu mengadaptasi *game* behavior-nya dalam merespon musuhnya, baik ketika sesi bermain *game* atau dalam diantara sesi tersebut. Dalam kasus tertentu, penskalaan tingkat kesulitan pada *game* dinamis menggunakan AAI untuk secara otomatis mengadaptasi parameter *game* dan *behavior* pada kondisi real time menurut tingkat keahlian pemain pada *game*.

D. K-Nearest Neighbor (K-NN)

Nearest Neighbor merupakan metode untuk mengklasifikasikan suatu data baru berdasarkan similaritas atau kemiripan dengan labeled data. Similaritas menggunakan metrik jarak dengan satuan *Euclidian*. *K-Nearest Neighbor* (KNN) adalah metode pengembangan dari NN (*Nearest Neighbor*). Dimana, algoritma ini termasuk ke dalam algoritma *supervised learning* dimana hasil dari *instance* yang baru diklasifikasikan berdasarkan mayoritas dari kategori K-tetangga terdekat. Tujuan algoritma ini adalah untuk mengklasifikasikan obyek baru berdasarkan atribut dan sampel dari data training. Algoritma KNN menggunakan *neighborhood classification* sebagai nilai prediksi dari nilai *instance* yang baru [14]. Metode ini bekerja berdasarkan jarak minimum dari data baru ke data *training sampel* untuk menentukan K tetangga terdekat. Selanjutnya kita dapatkan nilai mayoritas sebagai hasil prediksi dari data yang baru tersebut.

Dalam penelitian ini dikembangkan metode KNN dengan memberikan nilai bobot terhadap atribut yang digunakan [14]. Sebelum masuk pada metode KNN, *Player* melakukan aksi kepada *computer player* dan memasukkan aksi tersebut ke dalam dataset. Dataset tersebut akan dilakukan perhitungan dengan metode KNN untuk mengambil keputusan aksi apa yang akan diambil. Nilai pembobotan diberikan berdasarkan aksi yang diberikan oleh *player*. Berikut dijabarkan rancangan desain kecerdasan buatan menggunakan KNN berbobot.

1) Normalisasi Data

Sebelum dilakukan penentuan nilai bobot yang digunakan, dilakukan normalisasi data. Proses ini dilakukan untuk mengoptimalkan data yang diperoleh. Persamaan normalisasi data ditulis dengan menggunakan persamaan (1) sebagai berikut.

$$X_{norm} = \frac{X_i - X_{min}}{X_{max} - X_{min}} \quad (1)$$

Hasil yang didapat dari proses normalisasi di atas adalah mengubah semua nilai dari range 0 sampai 1.

2) Nilai Pembobotan

Nilai bobot yang diberikan berdasarkan aksi yang dilakukan oleh *player*. Pengaruh atribut pada tiap aksi ditunjukkan pada Tabel I.

Pada Tabel I tersebut ditunjukkan bahwa aksi *heal* berpengaruh dengan atribut *energy*, aksi *attack* berpengaruh pada atribut *energy* dan *damage*, aksi *defense* berpengaruh dengan atribut *health* dan *armor*, dan aksi

TABEL I
PENGARUH ATRIBUT PADA SETIAP AKSI

Aksi	Atribut
<i>Heal</i>	<i>Energy</i>
<i>Attack</i>	<i>Energy, Damage</i>
<i>Defense</i>	<i>Health, Armor</i>
<i>Ultimate</i>	<i>Energy, Damage</i>

ultimate berpengaruh pada atribut *energy* dan *damage*. Syarat dalam nilai pembobotan yaitu jika dijumlahkan hasilnya adalah 1. Nilai pembobotan yang dipakai dalam pertandingan ditentukan berdasarkan hasil uji coba sekaligus dilihat hasil jumlah kemenangan yang diperoleh. Jumlah kemenangan tersebut dijadikan sebagai titik acuan apakah AI sudah mampu beradaptasi dengan perilaku *player*. Untuk nilai pembobotan dari masing-masing atribut didapatkan dengan menggunakan persamaan (2) sebagai berikut.

$$\alpha = 1 - \beta \tag{2}$$

3) Menghitung Kedekatan

Setelah mendapatkan nilai data baru untuk semua data, selanjutnya dihitung jarak kedekatan antar data. Perhitungan tersebut menggunakan persamaan *euclidean distance* pada persamaan (3).

$$d_{ij} = \sqrt{\sum_{k=1}^n (a_{ik} - a_{jk})^2} \tag{3}$$

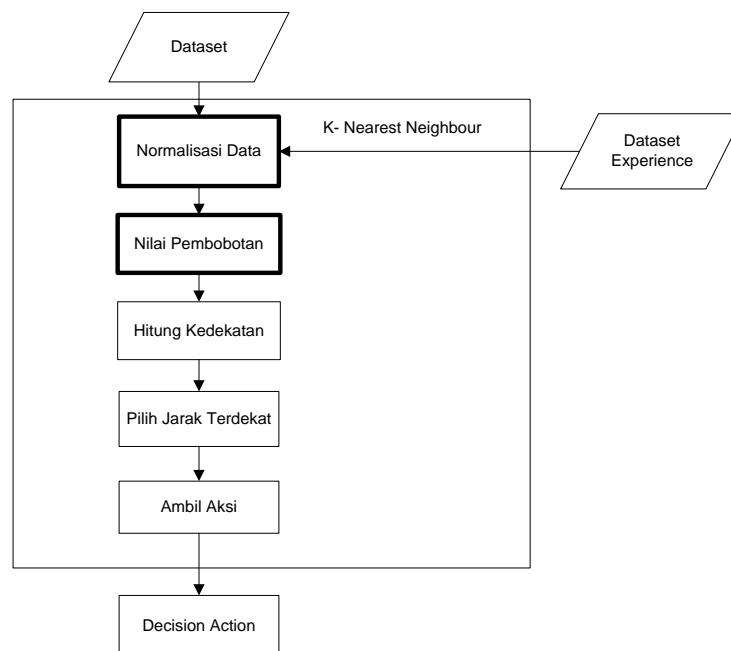
dimana n adalah jumlah atribut dan d_{ij} adalah jarak antar data. Semakin kecil nilai jarak yang didapat maka semakin mirip kedua data tersebut. Data yang dihitung nilai kedekatannya adalah data *training* (dataset *experience*) dan data testing (data kondisi AI terakhir).

4) Memilih Jarak Terdekat dan Ambil Aksi

Setelah dihitung semua nilai kedekatan antar data, selanjutnya akan dipilih nilai kedekatan dengan nilai yang terkecil. Nilai terdekat akan digunakan untuk menentukan keputusan AI yang akan diambil. Sehingga AI akan melakukan aksi diantara aksi yang tersedia pada *player*.

Aksi hanya ditentukan berdasarkan jarak terdekat/terkecil saja. Dimana, artinya K-NN hanya memiliki $K=1$. Dikarenakan *multi class* atau aksinya bervariasi. Proses K-NN akan diulangi terus-menerus sampai permainan berakhir.

Selanjutnya rancangan desain kecerdasan buatan atau AI menggunakan metode KNN berbobot ditunjukkan pada Gambar 1.



Gambar 1. Rancangan Desain Kecerdasan Buatan (AI) Menggunakan Metode K-NN Berbobot.

III. METODE PENELITIAN

A. Desain dan Implementasi

Tahap ini merupakan tahapan mengenai langkah-langkah dalam pembuatan dan pengembangan *game fighting* yang meliputi: analisis kebutuhan, desain sistem dan implementasi.

1) Analisis Kebutuhan

Tahap pertama yang dilakukan adalah melakukan analisis kebutuhan. Analisis kebutuhan adalah suatu tahap pengumpulan informasi yang dapat dijadikan sebagai acuan dalam pengembangan *game* pertarungan. Pengumpulan informasi berupa kebutuhan perangkat keras dan perangkat lunak yang digunakan, serta informasi mengenai materi yang akan dijadikan sebagai konten dalam pembuatan. Oleh karena pengumpulan kebutuhan akan dilakukan dengan cara observasi baik terhadap pengguna yang akan menjadi target penelitian, ahli dalam bidang yang bersangkutan maupun sumber-sumber relevan yang dapat dijadikan sebagai referensi dalam pengembangan *game* tersebut. Tahap ini dilakukan agar nantinya *game* pertarungan yang dikembangkan sesuai dengan harapan dan kebutuhan pengguna.

2) Desain Model Game

Tahap kedua adalah melakukan desain model *game* yang dibuat. Desain model *game* merupakan rancangan dari hasil analisis kebutuhan [1][8][10]. *Game* yang akan dibangun memiliki *genre* yang bertema pertarungan berskala kecil. Secara garis besar *game* pertarungan yang akan dibuat memiliki 2 buah karakter yang saling menyerang satu sebagai *player* dan satu sebagai *enemy*. Setiap karakter memiliki empat atribut yaitu *health*, *energy*, *armor*, dan *damage*. Selain atribut, setiap karakter juga memiliki empat kemampuan yaitu *attack*, *ultimate*, *defense*, dan *heal*. *Enemy* dalam *game* akan dilakukan oleh *computer player* kecerdasan buatan dalam mengambil keputusan. Permainan dinyatakan berakhir jika *health* dari salah satu pemain habis dan pemain dengan nilai *health* tersisa dinyatakan menang.

3) Desain Kecerdasan Buatan

Perancangan desain kecerdasan buatan merupakan bagian dari desain model *game*. Kecerdasan buatan *computer player* dalam permainan dibangun dengan menggunakan metode K-NN berbobot. Nilai bobot bersifat statis dimana nilai bobot didapatkan berdasarkan hasil uji coba sebanyak 5 kali pertandingan, nilai bobot yang akan dipakai adalah nilai bobot yang memiliki hasil *win rate* yang paling seimbang. Besar kecilnya nilai bobot atribut tergantung dari pengaruh aksi yang dilakukan oleh *player*. Jumlah rasio pembobotan yang digunakan bernilai 1 dari semua atribut yang diterapkan pada *player*.

B. Pengujian

1) Pembuatan Dataset

Dataset dipergunakan untuk klasifikasi sebagai data training [7]. Pada penelitian ini dataset digunakan untuk training metode K-NN berbobot. Dataset tersebut berisi nilai-nilai dari atribut dataset yaitu *health*, *energy*, *damage*, dan *armor*. Dataset dibangun dari percobaan *user* melawan metode kecerdasan buatan *rulebase*. Dimana *user* yang dipilih merupakan pakar dalam *game*. Pada setiap proses pengumpulan data, akan disimpan perlakuan *user* dalam menghadapi musuh didalam *game* pertarungan. Selanjutnya data akan diuji dengan menggunakan metode KNN untuk mengetahui apakah dataset tersebut layak digunakan dalam pengujian, karena dataset juga mempengaruhi hasil pertandingan.

2) Contoh Perhitungan KNN berbobot

Menentukan aksi yang akan diambil pada dataset ke-50. Dimana diketahui nilai masing-masing atribut yang ditunjukkan pada Tabel II.

Selanjutnya, untuk mendapatkan *decision* dilakukan proses *training* dengan cara menghitung jarak pada tiap data. Sebelumnya, data *training* dan data testing dilakukan normalisasi dengan persamaan (1) sehingga hasil normalisasi dapat ditunjukkan pada Tabel III. Selanjutnya dilakukan perhitungan untuk nilai K-NN dengan pembobotan. Proses perhitungan dapat dilihat pada Tabel IV.

Dari hasil perhitungan jarak, *decision* yang akan diambil untuk data ke-50 yaitu data yang memiliki hasil perhitungan jarak yang terkecil. Sehingga aksi yang akan dipilih oleh AI sesuai dengan dataset yang memiliki jarak terkecil.

3) Skenario Pengujian

Pada tahap uji coba terhadap metode yang diusulkan akan dilakukan 3 skenario uji coba. Pertama menguji cobakan secara langsung *player* dengan metode K-NN berbobot. Kedua, menguji cobakan secara langsung *player* dengan metode K-NN. Ketiga, menguji cobakan secara langsung *player* dengan metode random. Pada masing-masing uji coba yang dilakukan akan memperhatikan seberapa banyak aksi yang dilakukan pada masing-masing *player*. Setelah dilakukan pengujian, selanjutnya hasil dari aksi yang dilakukan antara *player*

TABEL II
CONTOH PERHITUNGAN DATASET KE 50

No.	Health	Energi	Damage	Armor	Decision
50	50	40	5	5	?

TABEL III
HASIL NORMALISASI

No.	Health	Energi	Damage	Armor	Decision
1	1	1	0	0	S
2	0.8	0.6	0	0	S
....					
50	0.12	0.4	0.6	0.5	?

TABEL IV
HASIL PERHITUNGAN JARAK PADA KNN DENGAN PEMBOBOTAN

No.	Proses Perhitungan
1.	$(1-0.12)^2 \cdot (1-0.19) + (1-0.4)^2 \cdot (1-0.31) + (0-0.6)^2 \cdot (1-0.19) + (0-0.5)^2 \cdot (1-0.31) = 1.23$
2.	$(0.8-0.12)^2 \cdot (1-0.19) + (0.6-0.4)^2 \cdot (1-0.31) + (0-0.6)^2 \cdot (1-0.19) + (0-0.5)^2 \cdot (1-0.31) = 0.87$
....	
50	Hasil = Jarak Terkecil

dan enemy akan dihitung seberapa banyak kemenangan dan kekalahan dari *player* terhadap enemy pada masing-masing skenario uji coba. Hal ini perlu dilakukan untuk dapat mengetahui seberapa besar persentase tingkat kecerdasan AI terhadap *player* pada *game* tersebut. Uji coba yang dilakukan pada masing-masing skenario pengujian sebanyak 50 kali percobaan.

4) Analisis Hasil

Evaluasi perhitungan persentase tingkat kecerdasan AI terhadap *player* diketahui dari perbandingan jumlah kekalahan dan kemenangan yang dihasilkan pada setiap skenario pengujian yang dilakukan. AI dianggap berhasil meningkatkan kecerdasan apabila jumlah kemenangan dan kekalahan sebanding. Sehingga untuk mengukur hal tersebut diusulkan persamaan sebagai berikut. Evaluasi perhitungan persentase mengikuti persamaan (4) apabila jumlah rasio kemenangan *player* terhadap AI lebih besar, sedangkan jika rasio kemenangan *player* terhadap AI lebih kecil digunakan persamaan (5).

$$\% \text{Persentase Tingkat Kecerdasan Learning AI} = \frac{\text{jumlah kekalahan}}{\text{jumlah kemenangan}} \times 100\% \quad (4)$$

$$\% \text{Persentase Tingkat Kecerdasan Learning AI} = \frac{\text{jumlah kemenangan}}{\text{jumlah kekalahan}} \times 100\% \quad (5)$$

Persentase tingkat kecerdasan AI didapatkan dengan membandingkan jumlah kemenangan dan kekalahan *player* terhadap AI dengan skala dari 0 sampai 100. Jika nilai persentase mendekati angka 0 maka dapat dikatakan bahwa tingkat kecerdasan *learning* AI pada *game* tersebut adalah rendah. Begitu juga sebaliknya jika nilai persentase mendekati angka 100 maka dapat dikatakan bahwa tingkat kecerdasan *learning* AI pada *game* tersebut adalah tinggi.

IV. HASIL DAN PEMBAHASAN

A. Hasil Desain dan Implementasi

Pada tahap ini akan dijelaskan mengenai hasil pembuatan dari *game* pertarungan.

1) Hasil Analisis Kebutuhan

Dua cara yang dilakukan untuk melakukan analisis kebutuhan tersebut. Pertama dilakukan observasi terhadap pihak yang dapat membantu dalam mendesain sekaligus membantu dalam pembuatan *game*. Observasi dilakukan terhadap pihak desainer *game* dan dilakukan dengan wawancara. Kedua, mencari referensi yang relevan terkait *game* pertarungan yang akan dibuat. Berdasarkan analisis kebutuhan yang telah dilakukan, didapatkan sejumlah hasil sebagai berikut:

- Desain dan tampilan antar muka pada *game* pertarungan disesuaikan dengan jenis *game* pertarungan RPG.
- Dua buah karakter yang akan diujikan memiliki atribut yang sama.
- Setiap metode kecerdasan buatan memungkinkan untuk dapat mengambil keputusan pada setiap kemungkinan yang akan terjadi.
- Perbandingan akan dilakukan terhadap tiga buah metode kecerdasan buatan.

2) Hasil Desain Sistem dan Implementasi

Desain sistem adalah tahap dimana hasil dari analisis kebutuhan yang telah diperoleh kemudian dijadikan sebagai acuan dari desain *game* pertarungan yang akan dikembangkan. Perangkat lunak yang digunakan sebagai alat bantu dalam mengimplementasikan sistem adalah sebagai berikut:

- a. Adobe Flash CS6 sebagai alat bantu penulisan kode program, pembuatan animasi dan pembuatan antar muka. Dalam hal ini *game* pertarungan yang dikembangkan dibuat dengan menggunakan bahasa pemrograman Action Script 2.
- b. XAMPP sebagai alat bantu pembuatan database.
- c. PhotoScape versi 3.7 sebagai alat bantu dalam pengolahan gambar bertipe bitmap.
- d. Adobe Illustrator CS6 sebagai alat bantu dalam pengolahan gambar bertipe vector.

Selain menggunakan bantuan perangkat lunak, pengembangan *game* pertarungan juga dibuat dengan bantuan perangkat keras yaitu seperangkat komputer dengan spesifikasi sistem Operasi Microsoft Windows 10 64bit, Processor Core i5 3.7 Ghz, RAM 8GB, Hardisk 1TB, dan VGA Nvidia 1050ti 4GB.

Berdasarkan hasil dari kebutuhan yang diperoleh maka didapatkan hasil dari pembuatan desain sistem *pada game fighting* sebagai berikut:

- a. Konten utama pada *game* adalah berupa permainan RPG. Desain permainan yang digunakan adalah tipe *fighting* dimana terdapat dua buah karakter yang saling menyerang yaitu, *player* dan *computer player* (AI).
- b. Setiap *computer player* memiliki atribut yang terdiri dari *health*, energi, *damage* dan *armor*. *Health* adalah nyawa dari *computer player*, jika *health* yang dimiliki habis maka *computer player* akan mati dan kalah. Energi adalah atribut yang digunakan memenuhi *cost* setiap keputusan dari *computer player*. *Armor* adalah atribut yang digunakan untuk mengurangi serangan musuh. *Damage* adalah atribut yang digunakan untuk menambah besaran serangan terhadap musuh. Atribut merupakan atribut penting dalam permainan yang telah disesuaikan dengan kebutuhan dalam *game* pertarungan yang dibuat. Karena pada dasarnya *gameplay* dari *game* yang dibangun hanya berupa *fighting* tanpa adanya status *player*, sehingga cukup hanya memperhatikan status bagaimana pemain menyerang (*attack*) dan bertahan (*defense*). *Cost damage* dan *armor* membutuhkan nilai sebesar 15. Angka tersebut dianggap sebagai nilai paling ideal yang digunakan sesuai dengan *game* pertarungan yang sudah ada sebelumnya. Jika perbandingan nilai *damage* dan *armor* lebih besar dari pada *health* dan *energy* maka *game* akan cepat selesai. Begitu juga sebaliknya. Jika perbandingan nilai *damage* dan *armor* lebih kecil dari pada *health* dan *energy* maka *game* akan terlalu lama selesai.
- c. Terdapat empat keputusan aksi yang dilakukan yaitu *ultimate*, *attack*, *defense*, *heal*. *Ultimate* adalah serangan ke musuh dengan besaran serangan lebih besar dari pada serangan *attack*. *Attack* adalah serangan ke musuh dengan besaran tertentu. *Defense* adalah keputusan untuk bertahan dari serangan musuh dengan mereduksi serangan musuh sebesar *armor* yang dimiliki. *Heal* adalah keputusan untuk menambah jumlah energi dengan besaran tertentu.
- d. Setiap *computer player* memiliki kecerdasan buatan yang berbeda untuk mengambil keputusan. Dalam hal ini kecerdasan buatan yang digunakan metode KNN berbobot (metode usulan), KNN dan *Rulebase*.
- e. Dataset digunakan sebagai data *training* kecerdasan buatan yang akan dipakai oleh kecerdasan buatan untuk mengambil keputusan.
- f. Konten utama pada *game* adalah berupa permainan pertarungan (*fighting*), dimana dalam permainan terdapat dua buah karakter yang saling menyerang, satu sebagai *player* dan satu sebagai *enemy*.
- g. *Enemy* dalam *game* akan dilakukan oleh *computer player* yang memiliki kecerdasan buatan yang berbeda untuk mengambil keputusan. Dalam hal ini kecerdasan buatan yang digunakan adalah KNN-berbobot (metode usulan), KNN tanpa bobot (metode yang memiliki relevansi dengan metode usulan), dan *random*.

Gambar 2 merupakan tampilan antar muka menu utama pada *game* pertarungan.

B. Pengujian

1) Hasil Pembuatan Dataset

Dataset yang digunakan di dalam *game* ini telah ditentukan oleh peneliti yang didapat berdasarkan hasil uji coba. Dataset yang digunakan sebanyak 132 data. Dataset yang dibuat dilakukan pengujian untuk melihat tingkat akurasi dari dataset yang akan digunakan. Pengujian dilakukan dengan menggunakan KNN dengan metode 10-cross validation menghasilkan akurasi sebesar 70,58%. Tabel V merupakan contoh dataset yang digunakan.



Gambar 2. Tampilan antar muka menu utama pada game pertarungan

2) Hasil Skenario Pengujian

a. Desain Skenario Uji Coba

Pengujian dilakukan sebanyak 3 skenario yakni:

1. Uji coba *player* dengan AI (KNN berbobot)
2. Uji coba *player* dengan AI (KNN tanpa bobot)
3. Uji coba *player* dengan AI (*random*)

Pengujian dilakukan sebanyak 3 skenario uji coba, dengan masing-masing pengujian dilakukan sebanyak 50 kali pertandingan. Banyaknya pertandingan yang dilakukan disesuaikan dengan tujuan yang ingin dicapai. Jumlah pertandingan ini sudah mencapai jumlah yang ideal untuk dapat melihat tingkat kecerdasan AI dari masing-masing pengujian. Sesuai dengan pengujian yang dilakukan, jika jumlah pertandingan yang dilakukan

TABEL V
CONTOH DATASET

No	Health 1	Health 2	Energy	Damage	Armor	Aksi
1.	200	200	100	10	10	U
2.	180	180	70	12	10	U
3.	160	156	40	14	10	A
4.	148	142	20	15	10	A
5.	135	127	0	15	11	H
6.	135	127	15	15	11	H
7.	135	127	30	15	11	U
8.	107	97	0	15	13	H
9.	107	97	15	15	13	H
10.	107	97	30	15	13	U
11.	79	67	0	15	15	H
12.	79	67	15	15	15	H
13.	79	67	30	15	15	U
14.	51	37	0	17	13	H
15.	51	37	15	17	13	H
16.	51	37	30	17	13	U
17.	23	3	0	19	11	H
18.	23	3	15	19	11	H
19.	23	3	30	19	11	A
20.	190	190	80	11	10	U

TABEL VI
UJI COBA PENENTUAN NILAI BOBOT SKENARIO 1

No	Win Rate	Sisa Health	Bobot				Aksi Enemy				Aksi Hero			
			DP	HP	AP	EP	U	A	D	H	U	A	D	H
1.	m	120	0,25	0,25	0,25	0,25	7	7	8	16	9	5	10	18
2.	m	68	0,25	0,25	0,25	0,25	2	6	10	17	8	10	2	24
3.	m	100	0,25	0,25	0,25	0,25	4	2	5	9	8	0	2	11
4.	m	78	0,25	0,25	0,25	0,25	3	5	9	13	6	7	2	16
5.	m	43	0,25	0,25	0,25	0,25	5	3	1	10	7	1	2	10
6.	m	90	0,25	0,25	0,25	0,25	4	9	2	25	10	9	0	27
7.	m	100	0,25	0,25	0,25	0,25	4	7	2	15	4	7	4	14
8.	m	33	0,25	0,25	0,25	0,25	2	10	9	20	4	8	10	20
9.	m	11	0,25	0,25	0,25	0,25	3	7	7	14	7	5	3	13
10.	m	68	0,25	0,25	0,25	0,25	4	6	14	21	5	9	9	23

TABEL VII
UJI COBA PENENTUAN NILAI BOBOT SKENARIO 2

No	Win Rate	Sisa Health	Bobot				Aksi Enemy				Aksi Hero			
			DP	HP	AP	EP	U	A	D	H	U	A	D	H
1.	k	0	0,28	0,22	0,28	0,22	5	15	10	31	8	15	6	35
2.	m	43	0,28	0,22	0,28	0,22	4	9	8	24	6	10	7	25
3.	m	50	0,28	0,22	0,28	0,22	3	9	8	17	7	7	1	18
4.	m	44	0,28	0,22	0,28	0,22	3	5	9	13	6	7	2	16
5.	m	5	0,28	0,22	0,28	0,22	5	9	8	14	8	3	2	26
6.	m	40	0,28	0,22	0,28	0,22	5	2	7	21	6	3	10	21
7.	k	0	0,28	0,22	0,28	0,22	4	8	10	20	5	6	8	20
8.	m	33	0,28	0,22	0,28	0,22	1	6	11	29	2	16	12	29
9.	m	11	0,28	0,22	0,28	0,22	3	7	7	14	7	5	3	13
10.	m	58	0,28	0,22	0,28	0,22	4	6	14	21	5	9	9	23

TABEL VIII
UJI COBA PENENTUAN NILAI BOBOT SKENARIO 3

No	Win Rate	Sisa Health	Bobot				Aksi Enemy				Aksi Hero			
			DP	HP	AP	EP	U	A	D	H	U	A	D	H
1.	m	38	0,31	0,19	0,31	0,19	3	11	13	28	1	12	13	29
2.	m	16	0,31	0,19	0,31	0,19	4	9	12	24	6	10	7	25
3.	m	5	0,31	0,19	0,31	0,19	5	5	20	26	6	6	10	27
4.	m	43	0,31	0,19	0,31	0,19	3	6	15	19	4	6	10	19
5.	m	51	0,31	0,19	0,31	0,19	7	11	15	28	9	13	4	28
6.	m	40	0,31	0,19	0,31	0,19	1	6	11	29	2	16	12	29
7.	k	0	0,31	0,19	0,31	0,19	4	8	10	20	3	9	8	20
8.	m	59	0,31	0,19	0,31	0,19	1	6	11	29	2	16	12	29
9.	m	28	0,31	0,19	0,31	0,19	3	11	13	28	1	12	13	29
10.	m	58	0,31	0,19	0,31	0,19	4	6	14	21	5	9	9	23

TABEL IX
UJI COBA PENENTUAN NILAI BOBOT SKENARIO 4

No	Win Rate	Sisa Health	Bobot				Aksi Enemy				Aksi Hero			
			DP	HP	AP	EP	U	A	D	H	U	A	D	H
1.	k	0	0,22	0,28	0,22	0,28	7	3	8	23	7	4	6	24
2.	k	0	0,22	0,28	0,22	0,28	8	0	2	12	7	1	1	12
3.	m	14	0,22	0,28	0,22	0,28	6	5	3	16	7	8	0	17
4.	m	30	0,22	0,28	0,22	0,28	6	8	1	16	6	9	2	19
5.	m	58	0,22	0,28	0,22	0,28	5	1	7	15	5	5	4	19
6.	k	0	0,22	0,28	0,22	0,28	7	14	7	30	5	10	12	29
7.	m	38	0,22	0,28	0,22	0,28	7	1	1	12	8	1	1	12
8.	m	23	0,22	0,28	0,22	0,28	3	8	8	21	5	8	6	21
9.	m	24	0,22	0,28	0,22	0,28	5	4	2	11	7	2	0	11
10.	m	6	0,22	0,28	0,22	0,28	8	5	4	17	10	4	0	18

TABEL X
UJI COBA PENENTUAN NILAI BOBOT SKENARIO 5

No	Win Rate	Sisa Health	Bobot				Aksi Enemy				Aksi Hero			
			DP	HP	AP	EP	U	A	D	H	U	A	D	H
1.	k	0	0,19	0,31	0,19	0,31	12	2	1	19	9	8	3	19
2.	k	0	0,19	0,31	0,19	0,31	14	2	1	20	13	5	0	19
3.	k	0	0,19	0,31	0,19	0,31	11	2	1	19	11	1	0	20
4.	m	17	0,19	0,31	0,19	0,31	7	3	2	16	9	2	1	14
5.	m	28	0,19	0,31	0,19	0,31	6	5	2	15	7	5	1	17
6.	k	0	0,19	0,31	0,19	0,31	5	5	1	14	4	4	5	15
7.	m	30	0,19	0,31	0,19	0,31	8	1	3	14	7	6	1	15
8.	k	0	0,19	0,31	0,19	0,31	6	5	2	17	4	5	7	16
9.	m	16	0,19	0,31	0,19	0,31	7	1	0	12	7	3	0	13
10.	m	1	0,19	0,31	0,19	0,31	9	0	0	14	9	1	0	14

TABEL XI
HASIL PENGUJIAN SKENARIO PERTAMA ANTARA PLAYER DENGAN K-NN BERBOBOT (METODE USULAN)

No	Win Rate	Sisa Health	Bobot				Aksi Enemy				Aksi Hero			
			DP	HP	AP	EP	U	A	D	H	U	A	D	H
1.	k	0	0,19	0,31	0,19	0,31	6	5	1	14	5	4	3	15
2.	k	0	0,19	0,31	0,19	0,31	6	5	2	17	4	5	7	16
3.	m	25	0,19	0,31	0,19	0,31	5	10	8	29	8	15	5	33
4.	m	68	0,19	0,31	0,19	0,31	4	8	4	14	8	2	1	13
5.	m	64	0,19	0,31	0,19	0,31	6	4	2	14	8	1	2	12
6.	m	31	0,19	0,31	0,19	0,31	8	4	4	13	10	4	0	14
7.	m	36	0,19	0,31	0,19	0,31	9	3	4	14	8	5	1	16
8.	k	0	0,19	0,31	0,19	0,31	10	8	4	27	7	10	14	29
9.	m	53	0,19	0,31	0,19	0,31	10	3	6	20	13	3	2	22
10.	m	10	0,19	0,31	0,19	0,31	6	8	11	29	3	12	11	28
11.	m	55	0,19	0,31	0,19	0,31	6	10	4	23	7	5	11	21
12.	m	8	0,19	0,31	0,19	0,31	6	10	8	27	9	6	8	27
13.	k	0	0,19	0,31	0,19	0,31	5	5	1	16	6	3	2	12
14.	m	1	0,19	0,31	0,19	0,31	5	9	9	24	6	9	7	25
15.	m	39	0,19	0,31	0,19	0,31	4	7	5	13	7	3	2	14
16.	m	24	0,19	0,31	0,19	0,31	4	5	6	13	6	4	2	13
17.	m	17	0,19	0,31	0,19	0,31	9	8	5	22	14	4	1	21
18.	m	10	0,19	0,31	0,19	0,31	4	7	4	12	7	5	1	12
19.	m	37	0,19	0,31	0,19	0,31	6	8	6	19	8	8	5	22
20.	k	0	0,19	0,31	0,19	0,31	5	6	1	13	7	3	0	12
21.	m	39	0,19	0,31	0,19	0,31	7	3	7	18	8	5	1	19
22.	k	0	0,19	0,31	0,19	0,31	7	7	4	17	8	4	6	16
No	Win Rate	Sisa Health	Bobot				Aksi Enemy				Aksi Hero			
			DP	HP	AP	EP	U	A	D	H	U	A	D	H
23.	k	0	0,19	0,31	0,19	0,31	7	1	2	10	5	3	1	10
24.	m	40	0,19	0,31	0,19	0,31	6	2	4	12	7	4	1	15
25.	m	79	0,19	0,31	0,19	0,31	4	6	4	11	9	2	2	13
26.	m	32	0,19	0,31	0,19	0,31	4	7	6	14	8	4	2	14
27.	m	52	0,19	0,31	0,19	0,31	4	10	9	21	6	7	8	22
28.	m	33	0,19	0,31	0,19	0,31	7	3	2	12	8	2	3	12
29.	k	0	0,19	0,31	0,19	0,31	7	7	0	20	8	6	1	19
30.	m	43	0,19	0,31	0,19	0,31	9	3	4	18	8	3	8	16
31.	m	39	0,19	0,31	0,19	0,31	7	8	4	18	8	5	5	17
32.	k	0	0,19	0,31	0,19	0,31	6	3	0	14	6	1	3	15
33.	m	10	0,19	0,31	0,19	0,31	5	10	11	30	4	13	1	28
34.	m	26	0,19	0,31	0,19	0,31	4	9	9	20	4	6	12	20
35.	k	0	0,19	0,31	0,19	0,31	8	3	3	17	5	8	4	17
36.	k	0	0,19	0,31	0,19	0,31	14	5	6	35	16	2	7	38
37.	m	10	0,19	0,31	0,19	0,31	5	11	5	24	6	6	12	22
38.	m	26	0,19	0,31	0,19	0,31	5	10	9	23	5	11	9	24
39.	k	0	0,19	0,31	0,19	0,31	6	7	8	22	5	8	9	21
40.	m	10	0,19	0,31	0,19	0,31	4	6	4	11	9	1	3	15
41.	m	26	0,19	0,31	0,19	0,31	2	14	12	28	7	14	4	30
42.	m	10	0,19	0,31	0,19	0,31	3	13	8	24	2	12	11	24
43.	k	0	0,19	0,31	0,19	0,31	9	11	13	38	7	15	14	38
44.	k	0	0,19	0,31	0,19	0,31	10	2	9	32	8	4	10	30
45.	k	0	0,19	0,31	0,19	0,31	7	4	4	16	6	5	1	17
46.	k	0	0,19	0,31	0,19	0,31	6	4	2	16	5	3	4	15
47.	m	22	0,19	0,31	0,19	0,31	7	3	5	14	9	3	0	13
48.	k	0	0,19	0,31	0,19	0,31	9	6	4	25	8	11	6	26
49.	m	26	0,19	0,31	0,19	0,31	6	5	6	15	9	3	1	14
50.	m	17	0,19	0,31	0,19	0,31	3	8	11	23	5	9	6	24

kurang dari 50 pertandingan maka belum terlihat hasil dari kecerdasan AI yang diterapkan pada *game* tersebut. Sehingga jumlah pengujian dilakukan sebanyak 50 kali pertandingan. Setiap skenario dilakukan untuk membandingkan tingkat keberhasilan metode yang diusulkan dengan metode lain yakni KNN dan random. Metode KNN digunakan untuk perbandingan uji coba dikarenakan metode yang diusulkan memiliki relevansi terhadap metode usulan. Sedangkan metode *random* merupakan metode yang umum digunakan untuk AI pada *game* pada umumnya.

Selama pengujian, *player* memainkan *game* yang telah diberikan dan akan dicatat aksi yang *player* lakukan. Aksi yang dicatat diantaranya jumlah menang, kalah, sisa *health*, dan pemilihan aksi pada *game*. Faktor yang mempengaruhi pengujian ialah *user experience player* dimana adanya hal tersebut akan mempengaruhi *skill player* dalam memenangkan permainan. Sehingga uji coba akan dilakukan terhadap 5 pemain yang memiliki *user experience* dalam *game* pertarungan. *Player* yang ikut serta dalam uji coba adalah orang yang sudah pernah

TABELXII
HASIL PENGUJIAN SKENARIO KEDUA ANTARA PLAYER DENGAN K-NN TANPA BERBOBOT

No	Win Rate	Sisa Health	Aksi Enemy				Aksi Hero			
			U	A	D	H	U	A	D	H
1.	m	54	6	3	0	10	7	3	0	10
2.	m	72	4	6	1	20	6	6	1	20
3.	k	0	3	9	2	25	9	10	5	24
4.	m	77	7	1	2	12	6	4	1	12
5.	m	49	2	4	3	22	3	5	3	22
6.	k	0	5	9	8	14	6	8	8	26
7.	m	57	6	8	11	23	3	9	11	29
8.	m	72	7	0	0	9	7	1	0	9
9.	m	38	4	3	0	19	8	3	0	19
10.	m	66	5	3	0	14	7	4	0	14
11.	k	0	9	11	7	20	9	11	5	24
12.	m	100	4	9	2	25	10	9	0	27
13.	m	44	1	7	4	15	6	4	3	15
14.	m	73	3	6	3	10	6	3	3	9
15.	m	47	6	4	12	18	5	6	2	20
16.	k	0	4	1	12	19	6	13	1	17
17.	k	0	8	9	14	24	8	12	6	13
18.	m	80	8	4	4	25	3	8	0	26
19.	m	46	3	6	8	10	9	8	9	12
20.	m	37	3	5	2	14	9	4	6	14
21.	m	83	3	9	4	16	2	5	3	16
22.	m	18	3	8	2	14	4	8	2	14
23.	m	60	2	10	2	16	3	8	5	15
24.	m	51	4	7	4	15	5	6	5	15
25.	m	78	3	6	1	11	5	5	1	11
26.	k	0	4	9	3	17	3	8	5	17
27.	m	126	4	7	2	15	4	7	4	14
28.	k	0	6	5	2	17	4	5	7	16
29.	k	0	6	9	14	28	3	12	13	14
No	Win Rate	Sisa Health	Aksi Enemy				Aksi Hero			
			U	A	U	A	U	A	U	A
30.	m	60	3	5	10	27	2	11	13	27
31.	m	63	1	6	11	29	2	16	12	29
32.	k	0	8	3	7	20	4	2	11	19
33.	m	24	3	9	1	15	2	11	1	15
34.	k	0	1	10	4	20	0	16	4	20
35.	m	35	5	8	6	23	4	9	7	23
36.	m	67	9	2	7	21	6	3	10	21
37.	k	0	5	9	4	22	3	8	8	21
38.	m	12	5	5	2	15	5	6	2	15
39.	k	0	8	5	2	19	3	7	7	17
40.	m	33	6	2	3	15	7	2	4	14
41.	m	8	7	0	0	10	7	1	0	10
42.	m	32	6	1	0	8	7	0	1	8
43.	m	41	6	2	1	11	7	1	2	11
44.	m	11	6	2	0	10	6	3	0	10
45.	m	63	3	5	3	10	8	1	1	12
46.	k	0	7	5	3	13	3	7	7	14
47.	m	34	3	6	3	11	6	4	0	11
48.	m	36	3	6	5	11	10	1	0	15
49.	k	0	5	5	1	11	3	6	1	11
50.	m	38	7	1	1	12	8	1	1	12

bermain *game* pertarungan dengan rentang umur antara 20-30 tahun dengan intensitas sering bermain *game* yang berbeda.

b. Penentuan Nilai Bobot Atribut dalam *Game*

Nilai atribut yang dipakai dalam pengujian didapatkan rentang bobot yang paling baik yaitu nilai dari range 0.19-0.31. Hal ini didapatkan berdasarkan hasil uji coba sebanyak 5 kali pada 5 orang *player* secara bergantian dimana setiap uji coba dilakukan sebanyak 10 kali pertandingan. Nilai tersebut nantinya akan digunakan pada langkah selanjutnya. Penentuan nilai atribut dilakukan secara manual dengan melihat perbandingan jumlah kemenangan dan kekalahan yang didapatkan.

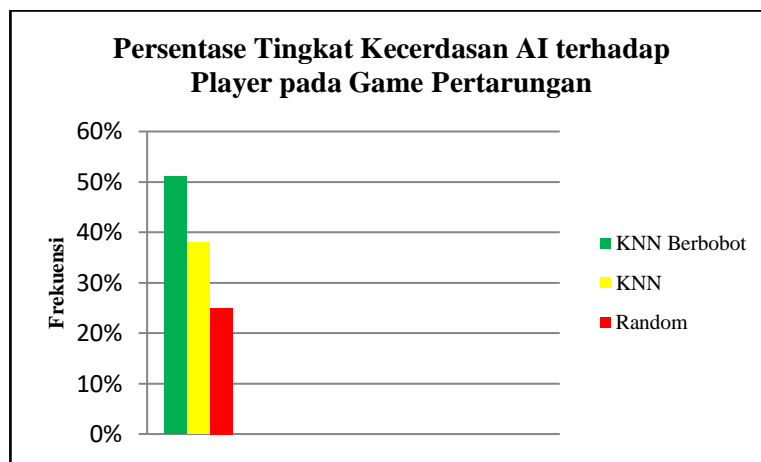
Pada uji coba pertama bobot pada masing-masing atribut bernilai sama, dimana nilai 1 dibagi dengan jumlah keseluruhan atribut yang ada. Selanjutnya, pada uji coba kedua nilai bobot atribut *health* dan *damage* bernilai lebih besar dari pada bobot atribut *armor* dan *energy*, yaitu sebesar 0.28 untuk atribut *health* dan *damage* dan 0.22 untuk atribut *armor* dan *energy*. Selanjutnya pada uji coba ketiga masih sama dengan skenario pada uji coba kedua, hanya saja nilai dari masing-masing atribut *health* dan *damage* dinaikkan sebesar 0.03 dan atribut

TABEL XIII
HASIL PENGUJIAN SKENARIO KEDUA ANTARA PLAYER DENGAN K-NN TANPA BERBOBOT

No	Win Rate	Sisa Health	Aksi Enemy				Aksi Hero			
			U	A	D	H	U	A	D	H
1.	m	60	5	3	11	16	5	8	4	19
2.	k	0	4	6	19	26	3	12	14	25
3.	m	74	4	7	18	25	2	16	11	26
4.	m	77	4	7	13	21	5	11	7	23
5.	m	49	3	9	9	19	5	11	4	21
6.	k	0	7	7	14	26	2	15	8	29
7.	m	57	4	6	11	19	4	13	3	21
8.	m	72	3	6	9	16	4	11	2	18
9.	k	0	3	14	1	19	4	7	8	17
10.	m	66	4	6	14	21	5	9	9	23
11.	m	121	2	7	14	16	10	5	20	12
12.	m	100	2	4	15	17	6	11	0	22
13.	m	44	4	6	4	13	7	5	1	15
14.	m	73	3	5	9	13	6	7	2	16
15.	m	47	2	9	8	16	5	11	1	19
16.	k	0	7	5	3	13	3	7	7	14
17.	k	0	8	3	8	19	5	9	4	20
18.	m	80	3	4	11	14	9	4	2	18
19.	m	46	5	4	9	15	6	8	2	18
20.	m	37	6	3	10	16	6	9	2	19
21.	m	100	3	3	8	10	9	2	0	14
22.	m	124	3	6	5	11	10	1	0	15
23.	m	140	3	5	3	10	8	1	1	12
24.	m	132	3	4	5	13	14	3	0	16
25.	m	168	1	6	5	12	10	1	0	15
26.	m	109	4	2	5	9	8	0	2	11
27.	m	126	5	1	7	15	16	0	4	19
28.	m	106	3	3	8	12	13	0	0	16
29.	m	132	2	3	10	9	10	0	1	14
30.	m	60	5	3	11	16	5	8	4	19
31.	k	0	6	7	8	22	5	8	9	21
32.	m	81	3	5	6	12	7	5	0	15
33.	m	75	1	7	9	12	7	6	0	17
34.	m	116	2	4	20	17	8	10	2	24
35.	m	20	3	7	15	22	8	11	3	26
36.	k	0	5	10	12	25	7	14	2	29
37.	m	73	2	6	4	10	6	5	0	12
No	Win Rate	Sisa Health	Aksi Enemy				Aksi Hero			
			U	A	U	A	U	A	U	A
38.	m	43	6	2	6	12	7	4	1	15
39.	m	68	4	5	10	16	7	7	3	19
40.	m	101	3	3	8	10	8	3	0	14
41.	m	83	1	7	11	13	7	8	0	18
42.	m	44	5	4	8	15	6	7	3	13
43.	m	83	2	6	5	9	7	3	1	12
44.	m	60	3	4	15	15	6	10	2	20
45.	m	39	3	8	13	19	5	11	6	22
46.	m	69	3	5	18	18	4	16	1	24
47.	k	0	6	6	17	25	2	16	9	27
48.	k	0	4	7	8	16	4	11	1	19
49.	m	78	2	5	6	11	8	3	0	14
50.	m	37	3	7	7	14	8	5	1	18

armor dan *energy* diturunkan sebesar 0.03 juga. Pada uji coba keempat nilai bobot atribut *health* dan *damage* bernilai lebih kecil dari pada bobot atribut *armor* dan *energy*, yaitu sebesar 0.22 untuk atribut *health* dan *damage* dan 0.28 untuk atribut *armor* dan *energy*. Kemudian pada uji coba kelima masih sama dengan skenario pada uji coba keempat, hanya saja nilai dari masing-masing atribut *health* dan *damage* diturunkan sebesar 0.03 dan atribut *armor* dan *energy* dinaikkan sebesar 0.03 juga. Pencarian bobot menggunakan perubahan rate 0.03 didasarkan pada jumlah atribut dan rancangan skenario. Sehingga nilai perubahan rate tersebut sudah dirasa cocok untuk mencari nilai bobot optimal tanpa membutuhkan kompleksitas yang cukup besar.

Berikut pada Tabel VI ditunjukkan hasil uji coba pertama, Tabel VII ditunjukkan hasil uji coba kedua, Tabel VIII ditunjukkan hasil uji coba ketiga, Tabel IX ditunjukkan hasil uji coba keempat, dan Tabel X ditunjukkan hasil uji coba kelima. Pada Tabel ditunjukkan hasil kemenangan dan kekalahan yang didapatkan pada setiap pertandingan yang ditunjukkan pada kolom *win rate*. Selanjutnya, dicatat pula sisa *health*, bobot masing-masing



Gambar 3. Grafik Persentase Tingkat Kecerdasan AI terhadap Player

atribut yang ditunjukkan pada kolom DP (*damage player*), HP (*health player*), AP (*armor player*), dan EP (*energy player*), dan aksi *enemy* serta *hero (player)* yang ditunjukkan pada kolom U (*ultimate*), A (*attack*), D (*defense*), dan H (*heal*).

c. Hasil Uji Coba Nilai Bobot Atribut Optimal

Dari hasil uji coba terhadap penentuan nilai bobot atribut didapatkan nilai bobot atribut sebagai berikut: DP (*damage player*) sebesar 0.22, HP (*health player*) sebesar 0.31, AP (*armor player*) sebesar 0.19, dan EP (*energy player*) sebesar 0.31.

Berdasarkan tabel hasil uji coba terhadap penentuan nilai bobot disimpulkan bahwa nilai bobot yang sesuai adalah skenario ke-5. Dimana rasio kemenangan dan kekalahan memiliki jumlah yang seimbang. Selanjutnya nilai bobot tersebut digunakan dalam pengujian tingkat kecerdasan AI.

d. Skenario Uji Coba 1 Antara Player dengan AI (KNN berbobot)

Pada Tabel XI ditunjukkan hasil pengujian skenario pertama yaitu antara *player* dengan AI (KNN berbobot). Pengujian tersebut dilakukan oleh 5 orang pemain secara bergantian dengan masing-masing pemain melakukan sebanyak 10 kali pertandingan. Dari tabel tersebut didapatkan hasil kemenangan *player* sebanyak 33 kali dan kekalahan sebanyak 17 kali. Setiap aksi yang diambil akan dilakukan perhitungan dengan mempertimbangkan nilai atribut dalam permainan. Pada Tabel ditunjukkan hasil kemenangan dan kekalahan yang didapatkan pada setiap pertandingan yang ditunjukkan pada kolom *win rate*. Selanjutnya, dicatat pula sisa *health*, bobot masing-masing atribut yang ditunjukkan pada kolom DP (*damage player*), HP (*health player*), AP (*armor player*), dan EP (*energy player*), dan jumlah aksi *enemy* serta *hero (player)* lakukan yang ditunjukkan pada kolom, U (*ultimate*), A (*attack*), D (*defense*), dan H (*heal*).

e. Skenario Uji Coba 2 Antara Player dengan AI (KNN tanpa bobot)

Pengujian selanjutnya dilakukan antara *player* dengan K-NN tanpa bobot. Dari hasil pengujian didapatkan jumlah kemenangan *player* sebanyak 36 kali dan kekalahan sebanyak 14 kali. Tabel XII menunjukkan hasil pengujian skenario kedua. Pada Tabel ditunjukkan hasil kemenangan dan kekalahan yang didapatkan pada setiap pertandingan yang ditunjukkan pada kolom *win rate*. Selanjutnya, dicatat pula sisa *health*, dan jumlah aksi *enemy* serta *hero (player)* lakukan yang ditunjukkan pada kolom, U (*ultimate*), A (*attack*), D (*defense*), dan H (*heal*).

f. Skenario Uji Coba 3 Antara Player dengan AI (random)

Pengujian terakhir dilakukan antara *player* dengan *random*. Dari hasil pengujian didapatkan jumlah kemenangan *player* sebanyak 41 kali dan kekalahan sebanyak 9 kali. Tabel XIII menunjukkan hasil pengujian skenario ketiga. Pada Tabel ditunjukkan hasil kemenangan dan kekalahan yang didapatkan pada setiap pertandingan yang ditunjukkan pada kolom *win rate*. Selanjutnya, dicatat pula sisa *health*, dan jumlah aksi *enemy* serta *hero (player)* lakukan yang ditunjukkan pada kolom, U (*ultimate*), A (*attack*), D (*defense*), dan H (*heal*).

3) Analisis Hasil Pengujian

Berdasarkan hasil pengujian yang telah dilakukan didapatkan hasil persentase dari rasio antara banyaknya kemenangan dan kekalahan yang didapat pada masing-masing skenario pengujian. Persentase tingkat kecerdasan AI didapatkan dengan menggunakan persamaan (4). Secara langsung hasil persentase tingkat kecerdasan AI terhadap *player* ditunjukkan pada Gambar 3. Perbandingan rasio kemenangan dan kekalahan yang dihasilkan untuk menunjukkan apakah AI sudah mampu beradaptasi (menyesuaikan) dengan perilaku *player*. Jika AI terlalu pintar maka sulit bagi *player* akan menang, begitu juga sebaliknya. Jika AI terlalu mudah *player* akan segera merasa bosan, hingga pada akhirnya *enjoyment* yang didapatkan pada *game* tersebut tidak ada. Pada

Gambar 3 bisa dilihat bahwa KNN berbobot mendapatkan nilai persentase sebesar 51%, dilanjutkan dengan KNN tanpa bobot sebesar 38%, dan *random* sebesar 25%. Dari ketiga metode yang diujikan, metode usulan yakni KNN berbobot memiliki persentase yang tinggi dibanding dengan 2 metode lainnya. Metode KNN berbobot merupakan metode yang memiliki titik berat pada pembobotan atribut *player*. Pembobotan yang diberikan pada masing-masing atribut *player* memberikan dampak terhadap keputusan yang diambil AI, sehingga AI dapat memberikan hasil yang optimal dalam pengambilan keputusan dalam memilih aksi. Namun, dalam hal ini metode KNN berbobot belum bisa memberikan hasil yang signifikan terhadap hasil tingkat kecerdasan AI. Pembobotan atribut pada metode KNN bersifat statis, artinya bobot bernilai sama dari awal sampai akhir pertandingan. Dengan adanya bobot yang bersifat statis menjadi salah satu faktor yang menyebabkan hasil dari tingkat kecerdasan KNN berbobot kurang optimal. Karena aksi yang dilakukan *player* pada setiap perbandingan berbeda, sehingga nilai bobot atribut pun seharusnya berubah mengikuti dari banyaknya aksi yang diambil oleh *player*.

V. KESIMPULAN

Penelitian ini hanya berfokus pada kecerdasan buatan atau AI pada *game* pertarungan. Metode usulan yang diterapkan yaitu KNN berbobot, dimana pada metode tersebut memberikan nilai bobot terhadap atribut yang digunakan dalam permainan. Berdasarkan hasil uji coba yang dilakukan pada penelitian ini dapat disimpulkan bahwa metode KNN berbobot mampu meningkatkan kecerdasan AI terhadap *player*. Dari analisa hasil pengujian bahwa KNN berbobot mendapatkan nilai persentase sebesar 51%, dilanjutkan dengan KNN tanpa bobot sebesar 38%, dan *random* sebesar 25%. Pembobotan atribut pada KNN berbobot memberikan pengaruh terhadap keputusan pengambilan aksi yang dilakukan oleh AI, sehingga hasil AI juga dapat memberikan hasil yang optimal. Namun, dalam hal ini metode KNN berbobot belum bisa memberikan hasil yang signifikan terhadap hasil tingkat kecerdasan AI. Pembobotan atribut pada metode KNN masih bersifat statis. Dengan adanya bobot yang bersifat statis tersebut menjadi salah satu faktor yang menyebabkan hasil dari tingkat kecerdasan KNN berbobot kurang optimal.

DAFTAR PUSTAKA

- [1] Brathwaite, B. and Schreiber, I. "Challenges for game designers". 1st edn. Rockland, MA, USA: Course Technology. 2009.
- [2] Emigh, M. S. *et al.* "Reinforcement Learning in Video Games using Nearest Neighbor Interpolation and Metric Learning", 32611(c), pp. 1–20. doi: 10.1109/TCIAIG.2014.2369345. 2014.
- [3] Li, J. and Kendall, G. "A hyper-heuristic methodology to generate adaptive strategies for games", *IEEE Transactions on Computational Intelligence and AI in Games*, PP(99), p. 1. doi: 10.1109/TCIAIG.2015.2394780. 2015.
- [4] Maroney, K. "My Entire Walking Life, The Games Journal | A Magazine About Boardgames". 2001.
- [5] Munajat, B. and Toto, A. "Implementasi Adaptive Artificial Intelligence pada game Capture The Flag dengan metode Dynamic Scripting", 1(1), pp. 695–705. 2014.
- [6] Nakamichi, T. and Ito, T. "Implementation and qualitative analysis of an adaptive computer Shogi program by producing seesaw game", *TAAI 2015 - 2015 Conference on Technologies and Applications of Artificial Intelligence*, pp. 453–460. doi: 10.1109/TAAI.2015.7407107. 2016.
- [7] Santoso, B. "Data Mining Teknik Pemanfaatan Data untuk Keperluan Bisnis". (1 ed.). Yogyakarta: Graha Ilmu. 2007.
- [8] Salen, K. and Zimmerman, E. Rules of Play: "Game Design Fundamentals, Nihon Ronen Igakkai zasshi. Japanese journal of geriatrics". doi: 10.1093/intimm/dxs150. 2004.
- [9] Thawonmas, R., Yamamoto, K. Mizuno, S. And Chu, Y. C. "Deduction of Fighting-Game Countermeasures Using the k-Nearest Neighbor Algorithm and a Game Simulator", pp. 0–4. 2015.
- [10] Urh, M. *et al.* "The Model for Introduction of Gamification into E-learning in Higher Education", *Procedia - Social and Behavioral Sciences*. Elsevier B.V., 197(February), pp. 388–397. doi: 10.1016/j.sbspro.2015.07.154. 2015.
- [11] Yannakakis, G. N. and Togelius, J. "A Panorama of Artificial and Computational Intelligence in Games", *IEEE Transactions on Computational Intelligence and AI in Games*, 7(4), pp. 317–335. doi: 10.1109/TCIAIG.2014.2339221. 2015.
- [12] Abdillah, F.I., Adams Jonemaro, M. E., dan Akbar, A. M.. "Implementasi Adaptive AI Pada Game Turn Based RPG Dengan Menggunakan Metode Hierirchal Dynamic Scripting". *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*. Vol.2, No.2. hal. 704. 2017.
- [13] S. Bedoya-rodriguez, C. Gomez-urbano, A. Uribe-quevedo, and C. Quintero, "Augmented Reality RPG Card-based Game", *Games Media Entertain. (GEM), IEEE*, pp. 3–6. 2014.
- [14] Zhan, Y., Dai, S., Mao, Q., Liu, L., and Sheng, W. "A Video Semantic Analysis Method Based on Kernel Discriminative Sparse Representation and Weighted KNN". *The Computer Journal. Computer and Communications Networks and Systems*. Vol. 58, No. 6. 2015.

Halaman ini sengaja dikosongkan