

**PENGEMBANGAN GAME CATUR DENGAN PYTHON: INTEGRASI
AI DAN GUI UNTUK PEMBELAJARAN INTERAKTIF**

**Fahmy Syahputra¹, Elsa Sabrina², Dinda³, Kristian Doktor A.
Panjaitan⁴, Rifaldi Alan Dahrís Harefa⁵, Sakinah Salsabila
Siregar⁶, Steven Sipahutar⁷**

Universitas Negeri Medan

E-mail: famybd@unimed.ac.id¹, elsasabrina@unimed.ac.id²,
dindaafritra@gmail.com³, kristianpanjaitan025@gmail.com⁴,
darisharefa56@gmail.com⁵, sakinah.siregar2004@gmail.com⁶,
floss.stephen@gmail.com⁷

Abstrak

Permainan catur adalah permainan strategis yang melibatkan dua pemain yang berusaha untuk mengalahkan lawan mereka dengan mematikan pergerakan lawan. Selain sebagai hiburan, catur juga mengajarkan pemain untuk berpikir secara abstrak dan menyelesaikan masalah, yang dapat memengaruhi pola pikir dalam kehidupan sehari-hari. Dengan pesatnya perkembangan teknologi digital, permainan catur juga mengalami transformasi, terutama dalam hal penerapan kecerdasan buatan (AI) dan antarmuka grafis pengguna (GUI). Tujuan dari penelitian ini adalah untuk mengevaluasi pengaruh tingkat kedalaman alpha-beta pruning terhadap kesulitan permainan dan waktu komputasi yang diperlukan. Algoritma ini diimplementasikan menggunakan bahasa pemrograman Python dalam lingkungan pengembangan Visual Studio Code (VS Code), dengan memanfaatkan pustaka Pygame untuk antarmuka grafis. Hasil penelitian menunjukkan bahwa meskipun algoritma alpha-beta pruning dapat mengoptimalkan pencarian dengan memangkas cabang yang tidak relevan. Temuan ini memberikan wawasan penting tentang bagaimana integrasi AI dan GUI dalam permainan catur. Dengan adanya AI yang mampu beradaptasi dengan gaya bermain individu, pemain dapat merasakan pengalaman belajar yang lebih personal dan mendalam, yang pada gilirannya dapat meningkatkan motivasi dan keterlibatan mereka dalam permainan. Pembelajaran interaktif ini tidak hanya terbatas pada aspek teknis permainan, tetapi juga mencakup pengembangan karakter dan kemampuan sosial, seperti kesabaran, disiplin, dan kemampuan untuk bekerja dalam tim saat bermain dalam format turnamen.

Kata Kunci: Alpha-Beta Pruning, Kecerdasan Buatan (AI), Antarmuka Grafis Pengguna (GUI), Catur, Algoritma Pencarian, Optimasi, Pembelajaran Interaktif.

1. PENDAHULUAN

Di era sekarang, masyarakat sangat menyukai permainan karena adanya target yang harus dicapai saat berpartisipasi dalam permainan. Orang cenderung mengamati, berpikir, memilih, dan berinovasi untuk mencapai tujuan mereka. Dengan bermain catur, Anda dapat meningkatkan keterampilan tersebut. Catur, sebuah permainan strategi yang dimainkan oleh dua orang, mengharuskan pemain mematikan gerakan lawan mereka. Ini telah lama dianggap sebagai alat yang bagus untuk mengasah kemampuan kognitif. Cacat, yang menuntut pemikiran logis, analitik, dan strategis, melatih kemampuan kognitif seperti ingatan, perhatian, dan berpikir kritis (Nandana Wahyu, 2024). Orang belajar berpikir abstrak dan menyelesaikan masalah saat bermain catur, yang penting untuk berbagai aspek kehidupan,

seperti pendidikan, karir, dan kehidupan sehari-hari.

Dalam era modern ini, permainan papan tetap menarik minat di kalangan generasi muda. Permainan papan merupakan jenis permainan yang umumnya melibatkan penggunaan papan, berbagai elemen seperti dadu, bidak, atau penanda, serta aturan yang mengatur jalannya permainan dan melibatkan setidaknya dua pemain. Seiring dengan kemajuan teknologi, permainan papan kini mengalami transformasi ke dalam bentuk digital.

Manfaat dari bermain game juga harus diperhatikan, seperti kecanduan game atau terlalu sering bermain game. Kecanduan bermain game dapat mengakibatkan kurangnya motivasi untuk melakukan aktivitas lain serta berpotensi merusak kesehatan mata akibat paparan layar ponsel atau komputer dalam waktu yang lama. Dampak negatif ini dapat diminimalkan dengan menetapkan jadwal bermain yang teratur, misalnya hanya bermain pada akhir pekan dan menghindari permainan pada hari-hari lainnya. Selain itu, teknologi kecerdasan buatan (AI) dalam game telah dikembangkan dengan algoritma khusus agar mampu meniru kecerdasan manusia (Ramadhana & Wahid, 2024). Sebagai contoh, algoritma minimax diterapkan dalam permainan papan seperti catur guna memungkinkan komputer memiliki kemampuan pengambilan keputusan layaknya pemain manusia.

Peneliti menerapkan algoritma Minimax sebagai bentuk kecerdasan buatan (Artificial Intelligence/AI) pada komputer yang berfungsi sebagai lawan dalam permainan. Algoritma Minimax terdiri dari dua konsep utama, yaitu MIN dan MAX. MIN mengacu pada peran komputer sebagai lawan yang berupaya memperoleh nilai serendah mungkin, sedangkan MAX merujuk pada pemain manusia yang berusaha mendapatkan nilai setinggi mungkin (Jayadi, Aziz, Rachman, & Yuliansyah, 2016).

Algoritma Minimax dirancang khusus untuk diterapkan dalam permainan papan seperti catur. Algoritma ini berfungsi dengan menganalisis setiap kemungkinan keadaan permainan pada setiap langkah serta menggunakan struktur pohon untuk merepresentasikan seluruh langkah yang dapat diambil selama permainan berlangsung. Dengan mengasumsikan bahwa lawan bermain secara optimal, algoritma ini berupaya menemukan strategi yang memaksimalkan keuntungan sekaligus meminimalkan potensi kerugian bagi pemain yang sedang menjalankan langkahnya. Dengan mempertimbangkan nilai maksimum dan minimum dari setiap kemungkinan langkah, algoritma Minimax mampu menentukan pilihan terbaik guna mencapai hasil optimal dalam permainan. Tulisan ini membahas penelitian yang bertujuan untuk membuat permainan Catur yang menantang untuk dimainkan pada papan Catur 8 x 8. Hasil penelitian menunjukkan bahwa mode random harus ditambahkan ketika komputer memilih Langkah (Husni Thamrin, 2023). Ini dilakukan agar komputer dapat bergerak lebih cepat tetapi pemain masih dapat mengalahkannya.

Salah satu cara utama AI mengubah catur adalah dengan menyediakan pengalaman belajar yang adaptif. Mesin catur dapat menyesuaikan diri dengan tingkat keahlian pemain, menawarkan tantangan yang sesuai untuk membuat mereka tetap terlibat dan termotivasi. Pendekatan adaptif ini selaras dengan konsep Zona Proksimal Perkembangan (ZPD) dari Vygotsky pada tahun 1978, yang menyatakan bahwa pembelajaran yang optimal terjadi ketika tantangan berada di luar kemampuan siswa saat ini, namun masih dapat dijangkau dengan bimbingan.

Meskipun integrasi kecerdasan buatan (AI) dalam pengembangan aplikasi seluler semakin menjadi perhatian, kajian literatur yang tersedia menunjukkan masih minimnya penelitian yang secara khusus membahas topik ini secara komprehensif. Penelitian terdahulu umumnya menyoroti aspek AI dan aplikasi seluler secara terpisah, namun belum banyak studi yang secara sistematis mensintesis perkembangan terbaru mengenai integrasi AI dalam desain

aplikasi seluler dalam konteks ekonomi digital (Muhammad Ibrahim, 2023). Oleh karena itu, penelitian ini bertujuan untuk mengisi kesenjangan tersebut dengan melakukan analisis mendalam, mengidentifikasi tren utama, tantangan, serta peluang yang ada, dan pada akhirnya berkontribusi terhadap wacana akademik dalam bidang yang semakin relevan ini.

Kecerdasan buatan (AI) menunjukkan bahwa robot dapat meniru manusia dalam berbicara, berpikir, belajar, merencanakan, dan memahami, yang dikenal sebagai kecerdasan mesin atau kecerdasan komputer. Python umumnya digunakan untuk AI oleh para ilmuwan data. Python berisi pustaka dan metode matematika bawaan yang memudahkan untuk memecahkan masalah matematika dan menganalisis data. Visualisasi data sangat penting dalam sistem perangkat lunak saat ini karena memungkinkan pengguna untuk mendapatkan wawasan dari kumpulan data yang sangat besar.

Stockfish adalah mesin catur terkuat yang diakui secara global, dikenal karena kemampuannya dalam menganalisis posisi dan menentukan langkah optimal. Dalam penelitian ini, kami menyelidiki kemampuan pemain catur untuk membedakan antara permainan melawan manusia dan mesin, dengan fokus pada Stockfish (Eisma, Y.B. dkk, 2024).

Sebanyak 24 pemain catur berpartisipasi dalam percobaan, memainkan delapan permainan Blitz melawan empat jenis lawan: manusia, Maia, Stockfish 16 pada level yang diturunkan, dan Stockfish 16 pada level maksimal. Hasil menunjukkan bahwa Stockfish pada level tertinggi biasanya diidentifikasi sebagai mesin, sementara gerakan dari level yang diturunkan sering dianggap 'aneh' oleh peserta.

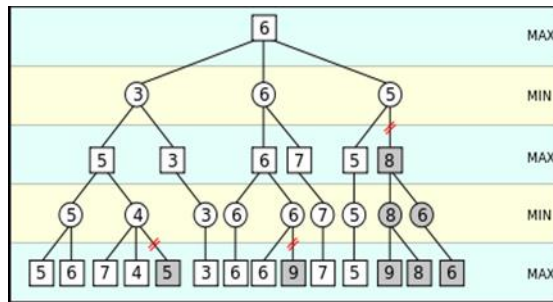
Dengan demikian, Stockfish tidak hanya meningkatkan keterampilan bermain catur, tetapi juga memberikan wawasan tentang interaksi antara manusia dan mesin dalam permainan catur.

Pada tahun 2008, peluncuran Stockfish menandai perkembangan signifikan dalam dunia catur komputer, menjadikannya salah satu program catur dengan kinerja terbaik hingga saat ini. Keberhasilan Stockfish tidak hanya berasal dari algoritma canggih, tetapi juga dukungan dari komunitas open-source yang dinamis, yang terus melakukan pembaruan dan perbaikan. Salah satu fitur inti Stockfish adalah penerapan teknik pendalaman iteratif, yang meningkatkan akurasi pemilihan gerakan dengan secara bertahap memperdalam pencarian. Selain itu, Stockfish menggunakan algoritma MVV-LVA untuk memprioritaskan penangkapan bidak, membantu menghindari tangkapan yang merugikan dan mengidentifikasi pertukaran yang menguntungkan. Algoritma Evaluasi Pertukaran Statis juga digunakan untuk menilai konsekuensi dari pertukaran bidak, memastikan fokus pada langkah-langkah yang menjanjikan.

Teknik pemangkasan, seperti pemangkasan gerakan nol, memungkinkan Stockfish untuk mempercepat proses pencarian dengan melewati gerakan yang tidak berpengaruh. Kombinasi berbagai teknik ini, bersama dengan pencarian paralel, menjadikan Stockfish salah satu komputer catur paling tangguh di dunia (Artur Side, 2024). Pembaruan signifikan juga terlihat dengan integrasi fungsi evaluasi berbasis jaringan saraf tiruan (NNUE), yang meningkatkan kemampuan Stockfish dalam mengevaluasi posisi secara akurat, terutama dalam situasi kompleks di mana metode tradisional mungkin kesulitan.

2. METODE PENELITIAN

Penelitian ini menerapkan algoritma pemangkasan alpha-beta guna meningkatkan efisiensi dalam pencarian langkah terbaik pada permainan catur. Pada Gambar 1, ditampilkan representasi pohon pencarian yang memanfaatkan algoritma alpha-beta pruning.



Gambar 1. Alpha-beta pruning

Seperti yang ditunjukkan dalam gambar diatas Alphabeta pruning sebagai berikut:

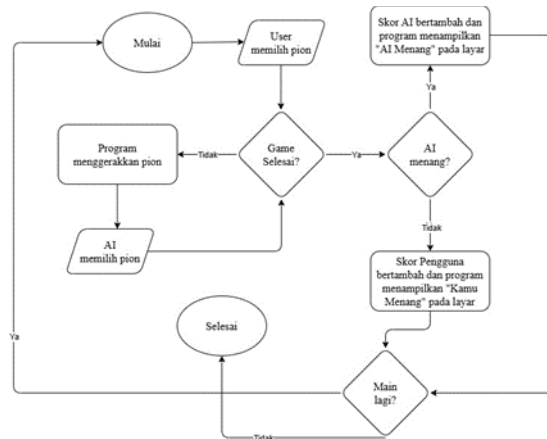
- Minimax MIN dan MAX dianggap sebagai dua pemain. Algoritma pruning alpha-beta merupakan metode pencarian yang dirancang untuk mengurangi jumlah nodus yang harus dievaluasi oleh algoritma minimax dalam struktur pohon pencarian. Pemain maximal (Max) dan pemain min (Min) masing-masing berusaha memaksimalkan skor.
- Alpha adalah kondisi nilai terbaik yang dapat dicapai oleh nilai Max, sedangkan Beta adalah kondisi skor maksimal yang dapat dicapai oleh pemain Min. Pruning alpha-beta digunakan dalam permainan catur untuk mengurangi jumlah posisi yang perlu dievaluasi.

Penelitian ini dimulai dengan pengembangan permainan Catur dengan ukuran papan 8 x 8. Dalam pengembangan ini, terdapat tiga mode permainan yang ditawarkan:

- Mode 0: Permainan antara dua manusia.
- Mode 1: Permainan antara manusia dan komputer, dimana komputer memilih langkah menggunakan algoritma minimax.
- Mode 2: Permainan antara manusia dan komputer, dimana komputer memilih langkah secara acak.

Pengamatan dilakukan selama permainan antara manusia melawan komputer. Setelah itu, penelitian dilanjutkan dengan pengembangan permainan catur dengan ukuran papan 8 x 8. Pada tahap awal, permainan ini juga dirancang dengan tiga mode yang sama seperti pada papan 3 x 3. Namun, karena waktu eksekusi algoritma minimax pada papan 8 x 8 terlalu lama, peneliti memutuskan untuk tidak menerapkan algoritma minimax secara penuh. Sebagai solusi, diperkenalkan mode 3, di mana komputer menentukan langkah secara acak pada tahap awal permainan sebelum beralih ke penggunaan algoritma minimax untuk langkah-langkah berikutnya. Permainan ini dikembangkan menggunakan bahasa pemrograman Python dengan memanfaatkan modul pygame. Alur permainan dapat diamati pada Gambar 1. Di awal permainan, pemain diberikan opsi untuk memilih mode permainan dengan menekan tombol 0, 1, 2, atau 3. Setelah mode dipilih, pemain dapat mengklik salah satu kotak kosong pada papan, yang kemudian akan terisi dengan tanda silang (marker). Jika permainan melibatkan komputer sebagai lawan, komputer akan menentukan kotak kosong yang akan diisi berdasarkan mode yang telah dipilih, baik dengan pemilihan acak maupun dengan menerapkan algoritma minimax.

Permainan berakhir ketika salah satu pemain atau komputer berhasil meraih kemenangan, atau ketika seluruh kotak telah terisi marker, yang menunjukkan bahwa permainan berakhir dengan hasil imbang. Kemenangan ditentukan jika seorang pemain atau komputer mampu menyusun marker dalam garis lurus secara berurutan sebanyak tiga atau empat kotak.



Gambar 2. Alur program game

Saat komputer menentukan kotak kosong untuk ditempati oleh marker, algoritma yang digunakan bersifat rekursif, yang berarti algoritma ini menganalisis seluruh kemungkinan penempatan marker pada kotak kosong yang tersedia. Implementasi algoritma minimax bertujuan untuk menemukan langkah optimal bagi komputer. Dalam prosesnya, setiap kemungkinan langkah dievaluasi dengan pemberian skor tertentu. Langkah yang berpotensi membawa kemenangan akan diberi nilai positif, sedangkan langkah yang dapat menyebabkan kekalahan akan diberi nilai negatif. Dengan cara ini, komputer dapat memilih langkah yang paling menguntungkan berdasarkan evaluasi skor tersebut, sehingga meningkatkan peluangnya untuk menang dalam permainan. Penelitian ini mengumpulkan data mengenai waktu yang diperlukan oleh komputer untuk menghitung satu langkah, yaitu saat mengisi tanda silang atau lingkaran ketika komputer mendapatkan giliran. Pengambilan data dilakukan khususnya ketika komputer menggunakan algoritma minimax untuk memilih langkah. Sebaliknya, pengambilan data tidak dilakukan saat komputer memilih langkah secara acak, karena proses tersebut diasumsikan berlangsung sangat cepat.

Data dikumpulkan untuk kedua jenis papan permainan, yaitu 8 x 8. Dapat diasumsikan bahwa waktu yang dibutuhkan untuk menentukan langkah terbaik akan lebih lama ketika masih terdapat banyak kotak kosong. Untuk setiap kotak kosong yang berbeda, data diambil sebanyak 5 kali untuk mendapatkan hasil yang lebih akurat.

Metode penelitian ini menggunakan Stockfish, sebuah program catur open-source yang dikenal sebagai salah satu komputer catur dengan kinerja terbaik. Keberhasilan Stockfish berasal dari algoritma canggih dan dukungan dari komunitas open-source yang aktif, yang terus melakukan pembaruan dan perbaikan. Dalam penelitian ini, Stockfish digunakan untuk menganalisis berbagai posisi catur dan mengevaluasi langkah-langkah yang diambil oleh pemain manusia.

Stockfish menerapkan teknik pendalaman iteratif untuk meningkatkan akurasi pemilihan gerakan. Teknik ini memungkinkan program untuk secara bertahap memperdalam pencarian, sehingga dapat mengalokasikan lebih banyak sumber daya komputasi pada lini permainan yang menjanjikan. Selain itu, Stockfish menggunakan algoritma MVV-LVA (Most Valuable Victim - Least Valuable Attacker) untuk memprioritaskan penangkapan bidak, yang membantu dalam menghindari tangkapan yang merugikan dan mengidentifikasi pertukaran yang menguntungkan.

Dalam hal evaluasi pertukaran, Stockfish menggunakan algoritma Evaluasi Pertukaran Statis untuk menilai konsekuensi dari pertukaran bidak. Algoritma ini memberikan "nilai tukar" untuk menentukan apakah pertukaran tersebut menguntungkan atau merugikan,

sehingga memastikan fokus pada langkah-langkah yang menjanjikan. Selain itu, Stockfish menerapkan teknik pemangkasan, seperti pemangkasan gerakan nol, yang memungkinkan program untuk melewati gerakan tertentu yang dianggap tidak berpengaruh terhadap hasil, mempercepat proses pencarian secara keseluruhan.

Pembaruan signifikan pada Stockfish juga terlihat dengan integrasi fungsi evaluasi berbasis jaringan syaraf tiruan. Fungsi ini meningkatkan kemampuan Stockfish dalam mengevaluasi posisi secara akurat, terutama dalam situasi kompleks di mana metode tradisional mungkin mengalami kesulitan.

Dalam percobaan ini, Stockfish digunakan dalam dua kondisi berbeda: Stockfish pada level tertinggi (StockfishH) dan pada level yang lebih rendah (StockfishL). Level StockfishL ditetapkan pada nilai rendah 4 dari 20 untuk meniru tingkat keterampilan manusia, sedangkan StockfishH beroperasi pada level maksimal 20 dari 20. Setiap permainan catur yang dimainkan oleh peserta dihadapkan pada kombinasi lawan manusia dan mesin, dengan gerakan lawan dilakukan setiap ± 10 detik dan urutan lawan diacak untuk meningkatkan kesulitan.

Setelah setiap permainan, analisis dilakukan untuk mengevaluasi langkah-langkah yang diambil oleh peserta. Evaluasi dilakukan dengan menggunakan algoritma Evaluasi Pertukaran Statis (SEE) dan teknik pemangkasan, termasuk pemangkasan gerakan nol, untuk mempercepat proses pencarian. Data yang dikumpulkan dari analisis permainan dievaluasi untuk memahami perbedaan strategi antara manusia dan mesin, memberikan wawasan yang lebih dalam tentang bagaimana pemain manusia membedakan antara gerakan mesin dan manusia dalam konteks permainan catur.

3. HASIL DAN PEMBAHASAN

Menurut H. J. R. Murray, penulis buku *History of Chess* (1913), permainan catur berasal dari India dan telah ada sejak abad ke-6. Di India, permainan ini dikenal dengan sebutan *chaturanga*, yang berarti "empat elemen yang terpisah." Pada awalnya, bidak catur hanya terdiri dari empat jenis. Dalam kepercayaan mistisisme India kuno, catur dianggap sebagai representasi dari alam semesta, sehingga sering dikaitkan dengan empat elemen kehidupan, yaitu api, udara, tanah, dan air. Hal ini disebabkan oleh sifat permainan catur yang dianggap mencerminkan berbagai aspek kehidupan manusia.

Catur berfungsi sebagai media untuk mengasah keterampilan keterampilan tersebut. Catur merupakan permainan strategi yang di dalamnya terdapat dua pemain dan mengharuskan mereka untuk mematikan gerak lawan mereka, telah lama dianggap sebagai alat yang sangat baik untuk mengasah kemampuan kognitif. Catur termasuk dalam keluarga permainan papan strategi. Permainan ini dimainkan antara dua pemain dan papanannya terdiri dari kotak-kotak berukuran 8x8 yang membentuk 64 kotak dengan warna terang dan gelap. Gambar 1 menunjukkan posisi awal permainan Catur. Setiap pemain memiliki 16 buah dengan kekuatan yang berbeda pada awal permainan.

Permainan catur dimainkan di papan 8x8 dengan 16 buah catur untuk setiap pemain. Tujuannya adalah mencapai skakmat, yaitu menyerang raja lawan tanpa bisa diselamatkan. Dalam game ini, pemain manusia dapat melawan AI yang menggunakan mesin catur Stockfish atau algoritma Minimax. Pada versi fisik, permainan ini dimainkan oleh dua orang, sedangkan dalam versi game komputer, salah satu pemain dapat digantikan oleh sistem komputer. Umumnya, pemain mencari tantangan dalam permainan, di mana mereka memiliki kesempatan untuk meraih kemenangan, tetapi juga menghadapi kemungkinan hasil imbang atau kekalahan.

1. Integrasi Stockfish

Integrasi Stockfish dalam permainan catur komputer memberikan dimensi baru dalam hal kecerdasan buatan. Stockfish adalah mesin catur open-source yang telah terbukti menjadi salah satu yang terkuat di dunia, berkat algoritma canggih dan pembaruan yang terus-menerus dari komunitas pengembang. Dengan menggunakan Stockfish, permainan catur dapat menjadi lebih menantang dan realistis bagi pemain manusia.

Stockfish diintegrasikan ke dalam game menggunakan library python-chess. Mesin ini menggunakan teknik seperti pendalaman iteratif dan algoritma MVV-LVA untuk menghasilkan gerakan strategis. Dalam pengujian, Stockfish mampu menghasilkan gerakan dalam waktu 2-5 detik pada tingkat kesulitan tinggi, dengan akurasi gerakan yang sangat baik.

Dalam penelitian ini, Stockfish digunakan untuk menganalisis langkah-langkah yang diambil oleh pemain manusia dan memberikan umpan balik yang berharga tentang strategi yang diterapkan. Dengan cara ini, pemain dapat belajar dari kesalahan mereka dan memahami bagaimana mesin berpikir, yang pada gilirannya dapat meningkatkan keterampilan bermain catur mereka. Integrasi Stockfish tidak hanya meningkatkan kecerdasan permainan tetapi juga memberikan wawasan yang lebih dalam tentang dinamika permainan catur.

2. Algoritma Minimax

Algoritma minimax adalah metode yang sangat efektif dalam permainan dua pemain, termasuk catur. Algoritma ini bekerja dengan mempertimbangkan semua kemungkinan langkah dari kedua pemain dan berusaha untuk memaksimalkan peluang kemenangan sambil meminimalkan risiko kekalahan. Dalam konteks permainan catur, minimax memungkinkan komputer untuk mengevaluasi berbagai posisi dan memilih langkah yang paling menguntungkan.

Dalam algoritma minimax, kedua pemain dikategorikan sebagai pemaksimal dan meminimal. Pemaksimal berusaha memperoleh skor tertinggi yang dapat dicapai, sedangkan meminimal bertujuan untuk mendapatkan skor serendah mungkin. Setiap kondisi papan permainan (board state) memiliki nilai tertentu yang terkait dengannya. Jika pada suatu kondisi pemaksimal berada dalam posisi menang, maka nilai tersebut umumnya bernilai positif, sedangkan jika meminimal yang menang, nilai yang diberikan biasanya bernilai negatif.

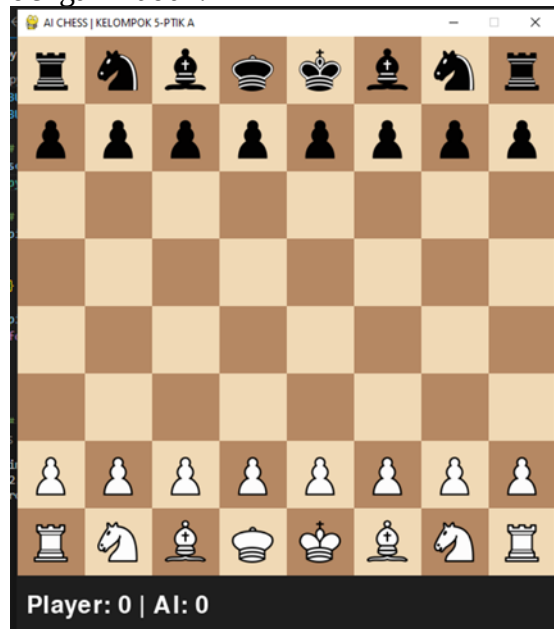
Namun, penggunaan algoritma minimax memiliki keterbatasan, terutama dalam hal waktu komputasi yang relatif lama. Ketika jumlah kotak kosong pada papan masih cukup banyak, proses eksekusi algoritma ini dapat memerlukan waktu yang signifikan, bahkan dalam hitungan beberapa detik untuk papan berukuran kecil. Untuk mengatasi kendala tersebut, teknik alpha-beta pruning dapat digunakan sebagai solusi. Teknik ini memungkinkan algoritma untuk mengabaikan posisi papan tertentu yang tidak mungkin diambil, berdasarkan probabilitas bahwa langkah tersebut tidak akan menguntungkan. Dengan menggunakan alpha-beta pruning, waktu eksekusi minimax dapat dikurangi secara signifikan, sehingga membuat permainan lebih responsif dan menyenangkan bagi pemain.

Dalam penelitian ini, kombinasi antara algoritma minimax dan teknik alpha-beta pruning digunakan untuk menciptakan pengalaman bermain yang lebih seimbang. Meskipun komputer tetap menjadi lawan yang kuat, penerapan langkah-langkah acak pada beberapa tahap permainan memberikan pemain manusia peluang untuk menang, menjadikan permainan lebih menarik dan menantang.

3. GUI dan Interaksi

Pengembangan antarmuka pengguna grafis (GUI) yang intuitif dan interaktif adalah

aspek penting dalam menciptakan pengalaman bermain yang menarik. Antarmuka grafis (GUI) yang dikembangkan menggunakan Pygame dirancang untuk meningkatkan kemudahan interaksi pemain dengan papan catur. Pemain dapat memilih gerakan dengan mouse, melihat analisis Stockfish secara real-time, dan mereset permainan kapan saja. Gambar 3 menunjukkan tampilan GUI game catur yang dikembangkan. Dalam penelitian ini, game catur berukuran 8 x 8 dikembangkan dengan GUI yang memungkinkan pemain untuk memilih langkah mereka dengan mudah.



Gambar 3. Tampilan utama game catur dengan papan 8x8 dan buah catur di posisi awal.



Gambar 4. Interaksi pemain dengan AI. Pemain memindahkan pion, setelah itu AI akan merespons.



Gambar 5. Notifikasi bahwa permainan telah berakhir, tombol untuk reset permainan dan skor masing-masing pemain

Interaksi yang baik antara pemain dan komputer sangat penting untuk meningkatkan keterlibatan pemain. GUI yang dirancang dengan baik tidak hanya memudahkan pemain dalam membuat keputusan, tetapi juga memberikan umpan balik yang jelas tentang langkah-langkah yang diambil oleh komputer. Dengan menampilkan analisis dari Stockfish, pemain dapat memahami mengapa langkah tertentu dipilih dan bagaimana strategi mereka dibandingkan dengan strategi mesin.

Selain itu, dalam penelitian ini, langkah-langkah awal komputer ditentukan secara acak untuk menjaga kecepatan permainan, sementara langkah-langkah selanjutnya menggunakan algoritma minimax. Pendekatan ini dirancang untuk menjaga keseimbangan antara tingkat kesulitan dan aspek hiburan, sehingga pemain memiliki kemungkinan untuk menang, bermain imbang, atau mengalami kekalahan. Dengan cara ini, permainan menjadi lebih menantang dan menarik dibandingkan dengan skenario di mana pemain selalu menang atau tidak pernah meraih kemenangan.

Dengan mengintegrasikan Stockfish, menerapkan algoritma minimax yang ditingkatkan, dan mengembangkan GUI yang interaktif, penelitian ini bertujuan untuk menciptakan pengalaman bermain catur yang lebih menantang dan memuaskan bagi pemain manusia. Pendekatan ini tidak hanya memperkuat kecerdasan dalam permainan, tetapi juga memberikan pemahaman yang lebih mendalam mengenai interaksi antara pemain manusia dan mesin dalam konteks permainan catur. Selain itu, dengan adanya fitur pembelajaran interaktif melalui permainan catur AI, pemain dapat secara proaktif memperbaiki pemikiran kritis, analisis, dan strategi untuk menghadapi tantangan AI. Berinteraksi secara langsung dengan sistem yang dapat menyesuaikan kesulitan permainan juga memberikan pengalaman belajar adaptif, di mana pemain secara bertahap dapat meningkatkan pemahaman mereka tentang konsep kecerdasan buatan, algoritma pemecahan masalah dan membuat keputusan lebih efektif. Oleh karena itu, permainan catur AI tidak hanya beroperasi seperti hiburan, tetapi juga pembelajaran kreatif dan alat pendidikan di dunia pendidikan.

4. Implementasi Teknis

a. Kode Program Utama

Kode utama mengontrol jalannya permainan, menangani input dari pengguna, dan memastikan permainan tetap berjalan.

- **main()**: Fungsi utama yang menjalankan loop permainan.
- **if __name__ == "__main__": main()**: Memastikan program berjalan hanya jika dieksekusi langsung.

Penjelasan:

- Program berjalan dalam loop while running yang terus menggambar papan dan bidak.
- Mendeteksi input dari pemain untuk memilih dan memindahkan bidak.
- Jika giliran AI, ia akan menghitung langkah dan menjalankannya.
- Memeriksa apakah permainan sudah selesai dan menampilkan hasilnya.

b. Kode untuk Integrasi Stockfish

Bagian ini bertanggung jawab untuk menghubungkan permainan dengan mesin catur Stockfish agar AI dapat bermain secara otomatis.

```
import chess
import chess.engine
engine_path = r"F:\nimed copy\S6\AI\GAME CATUR\stockfish\stockfish-windows-x86-64-avx2.exe"
engine = chess.engine.SimpleEngine.popen_uci(engine_path)
```

- **chess.Board()**: Membuat papan catur virtual.
- **chess.engine.SimpleEngine.popen_uci(engine_path)**: Menghubungkan ke Stockfish.
- **ai_move = engine.play(board, chess.engine.Limit(time=0.5))**: Memilih langkah terbaik berdasarkan batasan waktu 0.5 detik.

c. Kode untuk GUI Pygame

Bagian ini mengatur tampilan permainan menggunakan Pygame.

```
pygame.init()
screen = pygame.display.set_mode((WIDTH, HEIGHT))
pygame.display.set_caption("AI CHESS | KELOMPOK 5-PTIK A")
```

- **pygame.init()**: Menginisialisasi Pygame.
- **screen = pygame.display.set_mode((WIDTH, HEIGHT))**: Membuat jendela game.
- **pygame.display.set_caption()**: Mengatur judul jendela.

Fungsi utama dalam GUI:

- **draw_board()**: Menggambar papan catur.
- **draw_pieces()**: Menampilkan bidak di papan.
- **draw_score()**: Menampilkan skor di layar.

d. Kode untuk Animasi Gerakan

Bagian ini menangani animasi saat bidak berpindah tempat.

```
def animate_move(move):
```

```

start_col, start_row = chess.square_file(move.from_square), 7 -
chess.square_rank(move.from_square)
end_col, end_row = chess.square_file(move.to_square), 7 - chess.square_rank(move.to_square)
piece = board.piece_at(move.to_square)
if piece:
    img = piece_images[piece.symbol()]
    for i in range(10):
        draw_board()
        draw_pieces()
        x = start_col * SQUARE_SIZE + (end_col - start_col) * SQUARE_SIZE * i / 10
        y = start_row * SQUARE_SIZE + (end_row - start_row) * SQUARE_SIZE * i / 10
        screen.blit(img, (x, y))
        pygame.display.flip()
        time.sleep(0.015)

```

- Menggunakan loop untuk menggerakkan bidak secara bertahap.
- Menggambar ulang papan dan bidak setiap iterasi untuk menciptakan efek animasi.

e. Kode untuk Notifikasi Game Over & Tombol Main Lagi

Bagian ini menangani tampilan ketika permainan berakhir.

```

def draw_winner_text():
    font = pygame.font.Font(None, 50)
    text = font.render(winner, True, TEXT_COLOR)
    text_rect = text.get_rect(center=(WIDTH // 2, HEIGHT // 2 - 50))
    screen.blit(text, text_rect)

def draw_reset_button():
    font = pygame.font.Font(None, 50)
    button_rect = pygame.Rect(WIDTH // 3, HEIGHT // 2, WIDTH // 3, 50)
    mouse_pos = pygame.mouse.get_pos()
    color = BUTTON_HOVER if button_rect.collidepoint(mouse_pos) else BUTTON_COLOR
    pygame.draw.rect(screen, color, button_rect, border_radius=10)
    text = font.render("MAIN LAGI?", True, TEXT_COLOR)
    text_rect = text.get_rect(center=button_rect.center)
    screen.blit(text, text_rect)
    return button_rect

```

- **draw_winner_text()**: Menampilkan teks pemenang di tengah layar.
- **draw_reset_button()**: Menampilkan tombol “Main Lagi?” untuk mengulang permainan.
- Jika tombol ditekan, permainan **di-reset** ke awal.

f. Kode untuk Skor

Bagian ini menghitung skor pemain dan AI.

```

player_score = 0
ai_score = 0

def draw_score():
    font = pygame.font.Font(None, 40)
    score_text = font.render(f"Player: {player_score} | AI: {ai_score}", True, TEXT_COLOR)

```

```
screen.blit(score_text, (10, 620))
```

```
def check_game_over():  
    global game_over, winner, player_score, ai_score  
    if board.is_checkmate():  
        winner = "AI Menang" if board.turn else "Kamu Menang"  
        if board.turn:  
            ai_score += 1  
        else:  
            player_score += 1  
        game_over = True  
    elif board.is_stalemate() or board.is_insufficient_material():  
        winner = "Game Seri"  
        game_over = True
```

- **player_score** dan **ai_score** menyimpan jumlah kemenangan masing-masing pemain.
- **check_game_over()**:
 - Mengecek apakah ada **skakmat** atau **seri**.
 - Menambah skor berdasarkan pemenang.

4. KESIMPULAN

Penelitian ini telah berhasil mengembangkan permainan Catur dengan dua ukuran papan, yaitu 8 x 8. Dalam pengembangan ini, disediakan tiga mode permainan, yaitu mode manusia melawan manusia, mode manusia melawan komputer yang menentukan langkah secara acak (random), dan mode manusia melawan komputer yang menerapkan algoritma minimax. Pada mode random, komputer cenderung mudah dikalahkan oleh pemain manusia karena keputusan yang diambil tidak berdasarkan strategi yang terencana. Sebaliknya, ketika komputer menggunakan algoritma minimax, pemain manusia tidak pernah berhasil memenangkan permainan dan hanya dapat mencapai hasil imbang. Hal ini menunjukkan bahwa pengalaman bermain yang terlalu monoton, di mana pemain selalu menang atau selalu kalah, kurang menarik bagi mereka. Sebaliknya, pemain lebih menikmati permainan yang memberikan tantangan, dengan kemungkinan untuk menang, kalah, atau berakhir imbang.

Dengan mengombinasikan mode random dan mode minimax, permainan catur dapat memenuhi kriteria sebagai permainan yang menantang. Dalam konfigurasi ini, komputer dapat melakukan beberapa langkah secara acak di tahap awal sebelum beralih menggunakan algoritma minimax untuk langkah-langkah berikutnya. Pendekatan ini menghasilkan dinamika permainan yang lebih seimbang dan menarik, sehingga memberikan pemain pengalaman bermain yang lebih menantang.

Selain itu, penelitian ini juga menunjukkan bahwa integrasi game catur berbasis AI dalam pembelajaran interaktif dapat memberikan keuntungan yang signifikan untuk pengembangan kognitif pemain. Catur tidak hanya dapat meningkatkan keterampilan berpikir strategis dan analitis, tetapi juga dapat digunakan sebagai alat pembelajaran yang efektif untuk memahami konsep kecerdasan buatan, pengambilan keputusan berbasis algoritma, dan pemecahan masalah sistematis.

Menggunakan algoritma seperti minimax dan alpha beta, pemain dapat mempelajari bagaimana sistem AI mengevaluasi langkah-langkah yang mungkin dan mengembangkan

strategi yang paling optimal. Selain itu, interaksi langsung pemain catur dengan AI memungkinkan pemain untuk mengembangkan pemikiran adaptif, menganalisis pengambilan keputusan secara lebih kritis, dan memahami bagaimana teknologi dapat digunakan untuk menjadikannya tantangan yang rumit.

Oleh karena itu, permainan catur dasar AI dapat menjadi alat pendidikan yang menarik dan bermanfaat di dunia pendidikan, untuk menyajikan konsep dasar pemrograman dan kecerdasan buatan bagi siswa untuk meningkatkan keterampilan.

5. DAFTAR PUSTAKA

- AISHA LIAQAT, M. A. (2016). Metamorphic Testing of an Artificially Intelligent Chess Game. 4, 1-12
- Banjarnahor, P. J., Banjarnahor, J., Purba, W., Laia, Y., Sastra, E., Jijon, O., Sagala, R., Bastira, J., Okta, G. E., & Harmaja, J. (n.d.). PENGANTAR TEKNOLOGI INFORMASI.
- Dwi, G. M., Putri, I., Safitri, N., Norbertus, T., Suswanto, T., Putri, S., Sevtiyuni, E., Meiriza, A., & Citra, P. (2024). INTERAKSI MANUSIA DAN KOMPUTER. www.HeiPublishing.id
- Eisma, Y. B., Koerts, R., & de Winter, J. (2024). Turing tests in chess: An experiment revealing the role of human subjectivity. *Computers in Human Behavior Reports*, 16. <https://doi.org/10.1016/j.chbr.2024.100496>.
- Khatib Sulaiman, J., Thamrin, H., Wahid Aji Pamungkas, R., & Artikel Abstrak, I. (2023). Algoritma Minimax untuk Game Tic Tac Toe yang Menantang. *Indonesian Journal of Computer Science Attribution*, 12(2), 2023–1386.
- Machajewski, S. (2024, November 15). The AI Revolution in Chess and Its Impact On Education. Unlocking the joy of learning with a spritir of play .
- Maharaj, S., Polson, N., & Turk, A. (2022). Chess AI: Competing Paradigms for Machine Intelligence. *Entropy*, 24(4). <https://doi.org/10.3390/e24040550>
- Marthin Hokuta Kurniawan, D. U. (2020). Perbandingan Performa Algoritma Minimax Dan AlphaBeta Pruning Pada Game Catur Cina. *Jurnal Algoritma* , 1 (1), 102-110.
- Muhammad Farras Muttaqi1, K. B. (2018). Penerapan Algoritma Minimax Pada Permainan Tradisional Catur Jawa Skala 3x3 Untuk Meningkatkan Daya Tarik Pemain Skala 3x3 Untuk Meningkatkan Daya Tarik Pemain. *Jurnal Komputer dan informatika* , XX (No 2), 39-44.
- Muhammad Ibrahim, K., Kecerdasan Buatan dalam Desain Aplikasi Seluler, I., Ibrahim, M., Nasir, J., Komarudin, A., Maulana, A., & Hambali Akbar, M. (2023). Artikel Nusantara Computer and Design Review. *NCDR*, 1(1), 31–39. <https://journal.unusida.ac.id/index.php/ncdr/>
- Nandana Wahyu R, A. M. (2024). Penerapan Alpha-Beta Pruning dalam Pengembangan AI Catur: Evaluasi dan Peningkatan Strategi. *JAMASTIKA* , 2 (2), 30-45.
- Panji Novantara, M. N. (2017). Rancang Bangun Aplikasi Catur Menggunakan Algoritma Minimax Dengan Optimasi Alpha-Beta Pruning. *Cloud Informatika* , 3 (2), 1-15.
- Rais, A. H. (2018). Kecerdasan Buatan Permainan Catur dengan Minimax dan Alpha-Beta Pruning Sebuah Pendekatan Menggunakan Paradigma Depth-First Search dan Backtracking.
- Saltan, A. (2024). Systematic Analysis Of The Evolution Of Chess Computers . LUT

- University, 1-47.
- Side, A. (2024). Software and Systems Engineering Systematic Analysis of the Evolution of Chess Computers.
- Thoyyibah, Nanang, & Ratyanto, A. (2024). Algoritma Analisis Pada Javascript Dan Python. Purbalingga: Eureka Media Aksara.
- V Dhandayuthapani, B. (2024). Python Data Analysis and Visualization in Java GUI Applications Through TCP Socket Programming . Journal Information Technology and Computer Science , 16, 72-92.
- Wahid, R., Pamungkas, A., & Thamrin, H. (n.d.). PENERAPAN ALGORITMA MINIMAX PADA GAME TIC TAC TOE
- Wahyu, N. R., Mohammad Ziadhasya Rizqaarafi, A., Quthbul Widad, M., Puspita Sari, A., Pembangunan Nasional, U., timur, J., & Surabaya, K. (2024). Penerapan Alpha-Beta Pruning dalam Pengembangan AI Catur: Evaluasi dan Peningkatan Strategi. In JAMASTIKA (Vol. 3).
- Zhou, A. (2023). AlphaGambit: Parallelizing MiniMax for Chess.