

Penerapan HPA* Pada Game Ms. Pac-Man

Yoshua Aditya Kurnia¹, Eriq Muhammad Adams Jonemaro², Muhammad Aminul Akbar³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹yoshuaadityakurnia@gmail.com, ²eriq.adams@ub.ac.id, ³muhammad.aminul@ub.ac.id

Abstrak

Game Ms. Pac-Man merupakan permainan yang diminati beberapa peneliti dalam hal membuat kecerdasan buatan untuk memainkan Ms. Pac-Man. Berbeda dari Pac-Man, Ms. Pac-Man merupakan permainan non-deterministik sehingga banyak hal yang tidak pasti dalam permainan tersebut, membuat Ms. Pac-Man cukup sulit. Para peneliti menganggap hal tersebut merupakan tantangan yang menarik dalam perkembangan ilmu komputer khususnya kecerdasan buatan. Bahkan para peneliti membuat kompetisi untuk kecerdasan buatan terbaik dalam permainan ini, menggunakan skor sebagai perbandingan performa agen. Salah satu kompetisi yang ada adalah IEEE CIG 2011. Dalam permasalahan tersebut, penulis akan menggunakan metode Hierarchical Path-Finding A* (HPA*), metode sejenis A* untuk membuat agen kontroller pada Ms. Pac-Man. Tujuan dari penggunaan metode tersebut adalah menguji apakah metode HPA* dapat mencapai performa yang bagus. Performa pada metode A* belum mendekati skor tertinggi saat ini. Pada penelitian sebelumnya skor maksimal yang didapat menggunakan A* adalah 24640 dalam sepuluh kali percobaan. Dibandingkan dengan penelitian terbaru yang menggunakan decision tree dapat mencapai skor 43720, skor yang dicapai dengan metode A* kira-kira setengah dari penelitian terbaru. Untuk memperbaiki hasil metode A*, pada penelitian ini digunakan metode HPA*. Dengan membagi daerah permainan menjadi beberapa bagian melalui HPA*, skor yang didapat diharapkan lebih baik dari penelitian sebelumnya. Namun hasil yang didapat pada penelitian ini masih kalah, skor yang didapat adalah 17050. Oleh karena itu, pada penelitian ini dibahas implementasi HPA* beserta analisis performa dari metode HPA*.

Kata kunci: kecerdasan buatan, Ms. Pac-Man, HPA*, kontroler, path finding

Abstract

Ms. Pac-Man is a game used by the researchers to make an artificial intelligence for playing the game automatically. Since Ms. Pac-Man is a non-deterministic game, there are many things in the game that are random, for example the ghosts movement, making Ms. Pac-Man quite hard. Researchers think that as a interesting challenge for developing computer science especially artificial intelligence. Even more, researchers made a competition for the best artificial intelligence in this game, using score as a ranking for the agent performance. One of the competitions is IEEE CIG 2011. For this problem, the author are using Hierarchical Path-Finding A (HPA*) method, this method is an alternative from A* to make an agent controller in Ms. Pac-Man. The reason for choosing this method is because to test the HPA* method, whether it can achieve better result or worse. Agents using A* methods haven't reached the highest score in this game. In previous research, the maximum score achieved using A* method is 24640 in ten times testing. Compared to the newest research using decision tree method can reach a maximum score of 43.720. Roughly only half score that A* get in comparison to the newest research. To improve A* result, this research uses HPA* method. By dividing the game area into several areas using HPA*, the score will be expectedly better from the previous research. But in practice the results are not good, the final score is 17050. For this reason, this research implements the HPA* and analyze the performance of HPA* method.*

Keywords: artificial intelligence, Ms. Pac-Man, HPA*, controller, path finding

1. PENDAHULUAN

Ms. Pac-Man adalah sebuah arcade video

game yang diproduksi pada bulan Januari 1982 oleh Midway Manufacturing, penerbit dari Amerika Utara. Pada awalnya *Ms. Pac-Man*

merupakan permainan *Pac-Man* yang dibuat oleh programmer di General Computer Corporation dengan hasil perubahan bernama Crazy Otto. Proyek tersebut kemudian dibeli oleh Midway karena Namco belum merilis permainan *Pac-Man* yang baru sehingga Midway merilis game *Ms. Pac-Man*. Permainan ini cukup terkenal sehingga Namco, penerbit *Pac-Man* dari Jepang meresmikan judul permainan tersebut.

Ms. Pac-Man merupakan permainan yang diminati beberapa peneliti dalam hal membuat kecerdasan buatan untuk memainkan *Ms. Pac-Man*. Berbeda dari *Pac-Man*, *Ms. Pac-Man* merupakan permainan non-deterministik sehingga banyak hal yang tidak pasti dalam permainan tersebut, membuat *Ms. Pac-Man* cukup sulit. Para peneliti menganggap hal tersebut merupakan tantangan yang menarik dalam perkembangan ilmu komputer khususnya kecerdasan buatan (Lucas, 2011). Bahkan para peneliti membuat kompetisi untuk kecerdasan buatan terbaik dalam permainan ini, menggunakan skor sebagai perbandingan performa agen. Salah satu kompetisi yang ada adalah IEEE CIG 2011.

Dalam perkembangan agen yang sudah diimplementasikan untuk memainkan *Ms. Pac-Man*. Tan dan Teo (2016) telah mereview metode-metode yang telah diimplementasikan sebagai kecerdasan buatan untuk *Ms. Pac-Man*. Meskipun terdapat banyak metode, performa kecerdasan buatan dalam permainan ini masih jauh dibandingkan pemain manusia, dengan skor tertinggi 921360, sedangkan untuk kecerdasan buatan, skor tertinggi adalah 43720 (Foderaro et al, 2017) , Hal tersebut menunjukkan bahwa ilmu kecerdasan buatan dalam permainan ini masih bisa diteliti lebih lanjut untuk menemukan metode yang lebih baik dalam mencapai performa yang tinggi.

Dalam permasalahan ini akan digunakan metode Hierarchical Path-Finding A* (HPA*), metode sejenis A* untuk membuat agen pada *Ms. Pac-Man* karena HPA* mengecek kondisi secara bertahap (Botea et al, 2007). Pada A* jalur agen akan dibuat sampai ke tujuan, namun terkadang ada halangan baru yang muncul sehingga harus mengulang perhitungan pencarian jalur dan akhirnya perhitungan sebelumnya terbuang percuma. Performa pada metode A* belum mendekati skor tertinggi saat ini. Pada penelitian sebelumnya skor maksimal yang didapat menggunakan A* adalah 24640 dalam sepuluh kali percobaan. Dibandingkan

dengan penelitian terbaru yang menggunakan decision tree dapat mencapai skor 43720, skor yang dicapai dengan metode A* kira-kira setengah dari penelitian terbaru. Untuk memperbaiki hasil metode A*, pada penelitian ini digunakan metode HPA*. Dengan membagi daerah permainan menjadi beberapa bagian melalui HPA*, skor yang didapat diharapkan lebih baik dari penelitian sebelumnya.

Ms. Pac-Man adalah sebuah arcade video game yang dirilis pada Juli 1981. Dalam permainan ini terdapat karakter pemain yaitu *Ms. Pac-Man* yang berbentuk lingkaran kuning dengan pita pada kepalanya. Terdapat pil yang tersebar dalam sebuah labirin yang dijaga oleh empat hantu : blinky(hantu merah), inky(hantu biru), pinky(hantu pink), dan sue(hantu oranye). Pemain harus mengambil semua pil yang ada dalam labirin untuk mencapai tingkat berikutnya. Pemain tidak boleh menyentuh hantu pada umumnya karena jika pemain menyentuh hantu pemain dianggap mati dan nyawa pemain akan berkurang, namun jika pemain mengambil pil yang lebih besar, semua hantu akan berubah menjadi hantu biru tua yang dapat dimakan oleh pemain. Pemain akan mendapatkan skor saat memakan pil, hantu biru tua, atau buah. Untuk buah-buahan akan muncul pada kurun waktu dan bisa hilang juga. Bentuk labirin bisa berubah tergantung dari tingkatnya (Midway, 1982).

Dalam permainan ini agen dibuat untuk mengendalikan *Ms. Pac-Man* untuk memakan semua pill sambil menghindari hantu. Dari penelitian Ruck Thawonmas dan Hiroshi Matsumoto mereka menggunakan rule-based untuk mengatur tindakan selanjutnya yang diambil agen. Terdapat tujuh rule yang digunakan, dimana rule pertama mempunyai prioritas tertinggi, rule berikutnya mempunyai prioritas yang lebih rendah dari sebelumnya. Penulis menggunakan semua rule dari penelitian Thawonmas dan Matsumoto supaya perbandingan performa agen dapat lebih jelas, hanya berbeda dalam algoritme path finding dan deteksi objek permainan (Thawonmas, 2009).

Hierarchical Path-Finding A*(HPA*) merupakan perbaikan dari metode A*. Dalam permasalahan path-finding secara realtime untuk suatu permainan, komputer akan menggunakan banyak resource CPU dan memori, terutama jika ukuran daerah cukup besar. Dengan metode HPA*, sebuah peta daerah dimodelkan menjadi local cluster yang terhubung. Pada level local, jarak optimal untuk menyebrangi setiap cluster

sudah dihitung terlebih dulu dan disimpan pada cache. Pada level global, semua cluster ditelusuri secara lengkap (Botea et al, 2007).

2. METODOLOGI

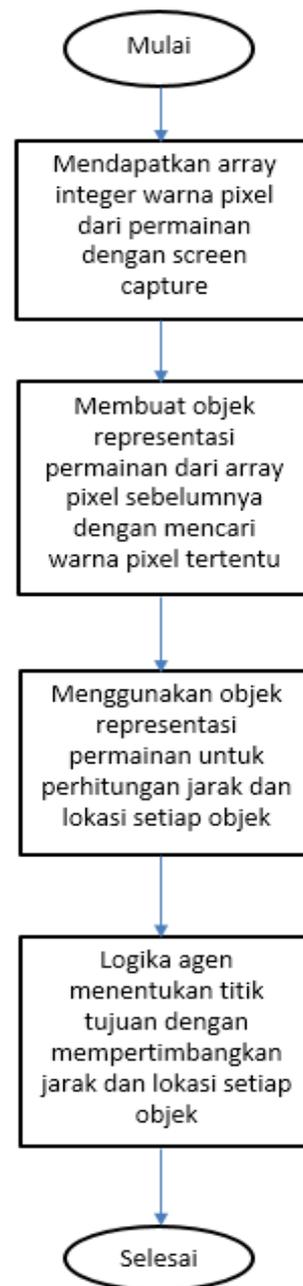
Metodologi penelitian menjelaskan langkah-langkah yang dilakukan dalam penelitian. Penelitian ini bersifat implementatif, dengan mengimplementasi algoritme HPA* sebagai path finding untuk agen. Metodologi penelitian yang digunakan terdapat pada Gambar 1:



Gambar 1. Metodologi Penelitian

3. PERANCANGAN

Bagian perancangan menjelaskan struktur program yang akan diimplementasikan. Terdapat tiga bagian yang perlu dirancang, yaitu metode screen-capture, pembuatan objek dari hasil screen-capture, dan logika agen. Untuk alur kerja program akan dijabarkan pada Gambar 2.



Gambar 2. Flowchart Program Agen

3.1. Screen-capture

Karena permainan Ms. Pac-Man dan program yang akan diimplementasikan merupakan proses yang berbeda, perlu adanya suatu modul yang mendapatkan gambar permainan secara real-time untuk program agen. Pada Gambar 3 adalah daerah yang diambil sebagai informasi untuk agen. Proses pengambilan gambar berdasarkan pixel, setiap pixel akan disimpan pada array pixel satu dimensi dengan ukuran 448 x 512. Setiap pixel memiliki nilai warna tersendiri, contohnya pixel hitam bernilai -16777216. Array pixel inilah

yang akan digunakan selanjutnya untuk membuat representasi objek permainan Ms. Pac-Man.



Gambar 3. Ukuran layar Ms. Pac-Man fullscreen, 448 x 512 pixel

3.2. Representasi Objek Permainan

Dalam permainan Ms. Pac-Man terdapat objek yang perlu dideteksi untuk diperhitungkan oleh agen dalam menentukan keputusan terakhir. Objek yang diprioritaskan adalah pil, hantu, dan Ms. Pac-Man. Untuk mendeteksi objek tersebut, dari array pixel sebelumnya dicek apakah warna pixel tersebut sama dengan pil, hantu, atau Ms. Pac-Man, jika warna pixel sama akan ditelusuri pixel yang bersebelahan dengan warna yang sama untuk menentukan ukuran objek tersebut dari pixel. Dari sumber source code yang dirujuk sebelumnya oleh penulis, objek tersebut diberi class dengan nama ConnectedSet. Setiap ConnectedSet memiliki variable warna, panjang, tinggi, koordinat x dan y. Dari objek tersebut digunakan oleh agen untuk mengetahui objek pada posisi tertentu merupakan hantu atau pil. Selain ketiga objek tersebut perlu ditambah objek *entrance* karena menggunakan algoritme HPA*. Cara menambahkan *entrance* adalah mengecek suatu posisi, dicek beberapa pixel di atas, kanan, bawah, dan kiri apakah pixel tersebut merupakan warna pil atau hitam. Jika memenuhi syarat tersebut akan dibuat objek ConnectedSet dengan warna yang disediakan untuk objek *entrance*.

Pada Gambar 4 merupakan gambar yang dibutuhkan untuk membuat objek representasi permainan. Gambar 4.a adalah Ms. Pac-Man sebagai agen, Gambar 4.b sampai dengan 4.e merupakan hantu sebagai halangan agen. Gambar 4.f adalah pill sebagai tujuan agen.

Untuk Gambar 4.g merupakan tujuan agen juga, namun mempunyai peran lain yaitu mengubah hantu menjadi hantu yang bisa dimakan seperti pada Gambar 4.h, dimana agen mempunyai prioritas untuk memakan objek tersebut. Gambar 5 menunjukkan jarak dari agen ke hantu sebagai hasil representasi objek permainan dengan memanfaatkan koordinat setiap objek. Jarak tersebut akan digunakan sebagai pengecekan kondisi pada logika agen.



Gambar 4. Objek yang dibutuhkan agen



Gambar 5. Garis ungu sebagai jarak antara agen dan hantu

3.3. Logika Agen

Penulis menggunakan metode HPA* sebagai path finding agen. Pada HPA* agen menentukan langkah berikutnya dengan cara melihat titik *entrance* terdekat pada posisi agen saat itu. Saat agen berhenti di titik *entrance*, akan dicari titik *entrance* berikutnya yang terdekat dengan tujuan. Jika jarak agen dengan tujuan sudah dekat, agen langsung menuju ke tujuan, mengabaikan *entrance* untuk sementara. Untuk menentukan cost dan titik tujuan agen, digunakan teori penelitian dari Thawonmas dan Matsumoto. Tabel 1 menjelaskan bagaimana cara kerja agen dalam menentukan titik tujuan dan cost agen.

Tabel 1. Pseudocode Logika Agen

No	Pseudocode
1	If rule 1 true
2	Agent stop moving to ambush ghost
3	Else if rule 2 true

```

4   Agent move to the closest
   power pill
5   isAStarVersionTwo ← true
6   Cost ← agentToPowerDistance
7   Else if rule 3 true
8   Agent move to the closest
   power pill
9   isAStarVersionOne ← true
10  Cost ← agentToPowerDistance
11  Else if rule 4 true
12  Agent move to the closest
   edible ghost
13  isAStarVersionOne ← true
14  Cost ← agentToEdibleDistance
15  Else if rule 5 true
16  Agent move to the closest
   pill
17  isAStarVersionOne ← true
18  Cost ← agentToPillDistance
19  Else if rule 6 true
20  Agent move to the closest
   edible ghost
21  isAStarVersionTwo ← true
22  Cost ← agentToEdibleDistance
23  Else if rule 7 true
24  Agent move to the closest
   pill
25  isAStarVersionTwo ← true
26  Cost ← agentToPillDistance
27  ENDIF
28  If cost > 32
29  Cost ← agentToEntrance * 1000
   Else
30  Cost ← Cost * 1000
31  ENDIF
32  IF isAStarVersionOne true
33  Cost ← Cost + 500000/
34  agentToEntrance ^ 2
   If isGoalCorner true
35  Cost ← Cost + 5000
36  ENDIF
37  If ghostNearbyGoal true
38  Cost ← Cost + 1000000
39  ENDIF
40  If ghostBehindAgent true
41  Cost ← Cost + 6000000
42  ENDIF
43  ENDIF
44  If Cost < bestCost
45  bestCost ← Cost
46  ENDIF
47

```

perbandingan mengikuti aturan lomba IEEE CEC 2009, dimana agen dijalankan pada permainan sebanyak sepuluh kali, lalu diambil skor rata-rata dan skor tertinggi agen. Agen yang memiliki skor lebih tinggi dapat disimpulkan agen tersebut mempunyai performa yang lebih bagus. Untuk aturan nyawa Ms. Pac-Man, nyawa awal adalah tiga dan nyawa tambahan pada skor kelipatan sepuluh ribu. Agen penulis akan dibandingkan dengan agen Thawonmas dan Matsumoto dimana performa agen tersebut diambil dari lomba IEEE CEC 2009.

4.2. Hasil Pengujian

Tabel 2 merupakan hasil pengujian agen berdasarkan skenario pengujian. Terdapat Gambar 6 sebagai grafik dari performa kedua agen. Gambar 7 merupakan layar dari program penulis sebagai pemantau cara kerja agen. Gambar 8 adalah hasil agen penulis pada percobaan ke-10.

Tabel 2. Hasil Pengujian Agen

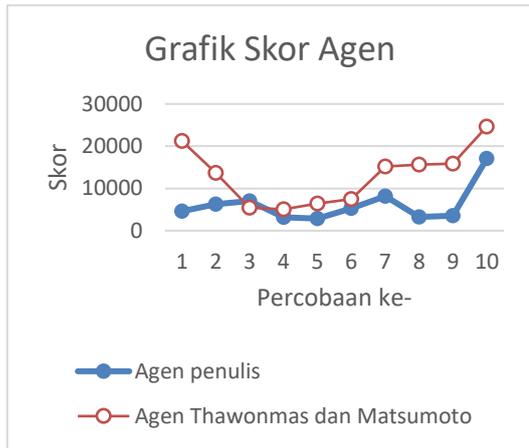
Percobaan ke-	Skor Akhir Penulis	Skor Akhir Thawonmas dan Matsumoto
1	4580	21200
2	6270	13650
3	7050	5470
4	3150	5070
5	2860	6430
6	5320	7450
7	8170	15170
8	3220	15630
9	3580	15880
10	17050	24640
Rata-rata	6125	13059

Sumber skor Thawonmas dan Matsumoto : <http://cswww.essex.ac.uk/staff/sml/pacman/CEC2009Results.html>

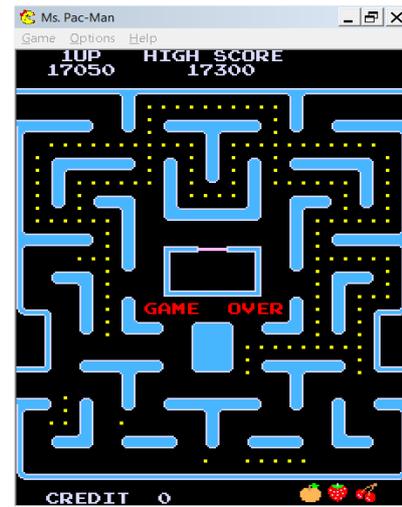
4. PENGUJIAN DAN ANALISIS

4.1. Skenario Pengujian

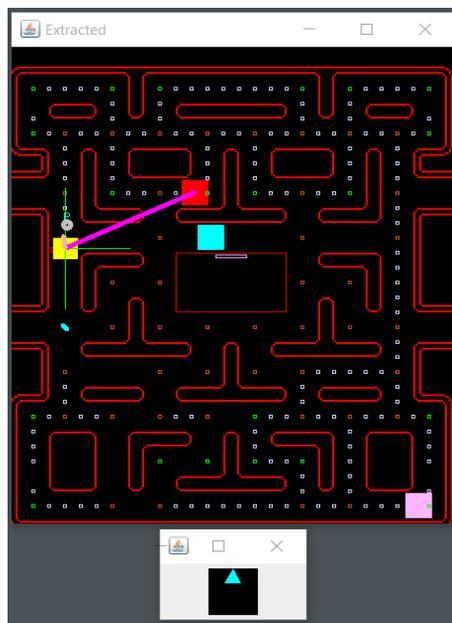
Untuk menguji performa agen, penulis membandingkan skor dari kedua agen. Aturan



Gambar 6. Grafik Skor Agen



Gambar 8. Percobaan ke-10 agen penulis



Gambar 7. Screen capture program

4.3. Analisis Hasil Pengujian

Dari sepuluh kali percobaan, rata-rata skor yang didapat adalah 6125. Agen penulis masih belum konsisten untuk menembus level pertama, ditandai dengan tujuh percobaan berakhir pada level pertama, sedangkan dua percobaan sampai pada level kedua, dan satu percobaan sampai level ketiga. Untuk perbedaan skor yang jauh pada percobaan ke-10 terjadi karena pemain mendapatkan nyawa tambahan setiap skor mencapai 10000. Selain itu posisi hantu lebih sering berkelompok sehingga saat memakan power pill dapat mencapai skor yang tinggi dengan memakan keempat hantu tersebut, skor bertambah 200, 400, 800, dan 1600 jika pemain memakan keempat hantu dalam satu durasi power pill.

Namun performa yang dicapai agen milik penulis masih kalah dengan agen milik Thawonmas dan Matsumoto. Hal ini dikarenakan terdapat beberapa masalah baru saat percobaan agen milik penulis. Selain itu kedua agen memiliki cara kerja yang berbeda. Berikut adalah beberapa faktor yang ditemukan penulis yang dapat mempengaruhi performa agen:

1. Metode screen capture tidak sepenuhnya dapat menginterpretasi objek dalam permainan dengan benar. Salah satu contoh kasus adalah saat hantu yang dapat dimakan bersebelahan seperti pada Gambar 9. Karena pada class ConnectedSet mendefinisikan objek berdasarkan panjang dan tinggi objek, hantu yang bersebelahan dianggap satu objek dengan ukuran yang berbeda dengan

definisi pada ConnectedSet, sehingga objek tersebut tidak dianggap hantu yang bisa dimakan oleh agen.



Gambar 9. Saat hantu bersebelahan, agen menganggap sebagai objek lain

2. Terdapat bug milik agen penulis saat mendekati power pill yang disebelah kiri bawah, agen tidak menunggu hantu lain mendekat terlebih dahulu, melainkan agen langsung memakan power pill tersebut. Skenario ini terjadi jika agen mendekati power pill dari sisi atas. Untuk power pill lainnya tidak ada masalah dengan penerapan rule 1.
3. Nilai konstan yang belum dioptimasi. Maksud dari pernyataan tersebut adalah pada koding program agen penulis, terdapat beberapa konstanta sebagai persyaratan dalam kondisi if. Penulis hanya mengestimasi nilai konstanta yang cocok untuk suatu kondisi. Terdapat beberapa kemungkinan untuk nilai konstanta yang lebih baik, Contohnya pada perhitungan cost hantu, penulis mendeteksi hantu yang dekat dengan titik *entrance* dengan nilai 48 pixel. Nilai tersebut didapatkan secara uji coba saja untuk menemukan hasil yang lebih bagus. Secara kondisi permainan 48 pixel setara dengan jarak tiga pill.
4. Waktu jeda dalam memproses informasi. Karena memproses logika agen secara real-time merupakan proses yang berat, diperlukan waktu jeda supaya program tidak keberatan dalam menjalankan tugasnya. Agen penulis diberikan waktu jeda sebesar dua millisecond, setelah agen memberi masukan untuk arah jalan agen. Penulis pernah mencoba untuk waktu jeda sebesar lima millisecond, namun terdapat masalah baru yang dialami oleh agen, contohnya saat agen menabrak tembok, terdapat waktu berhenti yang cukup terlihat, sehingga saat agen sudah dekat dengan hantu, agen dapat tertangkap oleh hantu. Untuk nilai yang optimal dalam waktu jeda juga masih dari uji coba penulis.
5. Metode image processing dengan pixel. Memproses gambar dengan pixel memang memberikan hasil yang cukup jelas

dengan informasi objek pada permainan, namun pada hal logika agen justru membuat kerugian karena objek permainan pada Ms. Pac-Man mempunyai panjang dan tinggi yang berbeda untuk Ms. Pac-Man dan hantu. Panjang dan tinggi objek tersebut bisa berubah saat berganti arah, misal saat Ms. Pac-Man bergerak ke atas, ukuran panjang dan tinggi adalah 12 dan 11 pixel. Sedangkan saat Ms. Pac-Man bergerak ke kanan, ukuran panjang dan tinggi menjadi 11 dan 12. Hal ini berpengaruh dalam menentukan titik tengah Ms. Pac-Man, karena berubah-ubah, akhirnya sulit untuk dipakai sebagai referensi dalam menentukan keputusan agen. Selain itu terdapat beberapa objek yang memiliki warna pixel sama, contohnya pada Gambar 10 untuk warna merah pada pita, hantu merah, buah, dan tembok merah. Karena agen penulis tidak mempertimbangkan objek buah, hasil screen capture dapat menjadi tidak akurat, terkadang buah tersebut dianggap halangan oleh agen.



Gambar 10. Warna pita, hantu merah, buah, dan tembok luar sama

6. Hantu yang tembus dari sisi kiri ke sisi kanan dan sebaliknya tidak dapat dideteksi oleh agen penulis, sehingga agen menjadi kehilangan titik tujuan. Karena pada saat kondisi tersebut objek hantu tidak terdeteksi, akhirnya agen mengikuti rule yang memakan pil terdekat.

5. KESIMPULAN

Berikut adalah kesimpulan penelitian untuk menjawab rumusan masalah yang telah didefinisikan:

1. Untuk implementasi metode HPA* pada agen Ms. Pac-Man, perlu dibuat titik *entrance* dengan cara mengecek suatu posisi dan objek disekitar posisi tersebut. Selanjutnya agen memperhitungkan jarak agen terhadap titik *entrance* terdekat.
2. Cara untuk mengukur performa agen adalah dengan menguji agen memainkan Ms. Pac-Man sebanyak sepuluh kali,

dengan nyawa awal tiga dan nyawa tambahan pada skor kelipatan sepuluh ribu. Cara tersebut mengikuti aturan lomba agen Ms. Pac-Man pada IEEE CIG 2011.

3. Performa agen penulis dengan metode HPA* tidak sebagus performa agen milik Thawonmas dan Matsumoto. Terdapat beberapa faktor yang dicantumkan oleh penulis yang dapat mempengaruhi performa agen dalam melakukan penelitian ini. Menurut penulis performa tersebut masih dapat diperbaiki dengan mencari solusi lain untuk masalah yang ditemukan oleh penulis.

DAFTAR PUSTAKA

- Botea, A. et al. (2007, Oktober). Near Optimal Hierarchical Path-Finding. Department of Computing Science, University of Alberta.
- Foderaro, G. et al. (2016, Januari). A Model-Based Approach to Optimizing Ms. Pac-Man Game Strategies in Real Time. IEEE.
- Lucas, S. M., 2009. Ms Pac-Man Competition IEEE CEC 2009 Results. [online] <<http://cswww.essex.ac.uk/staff/sml/pacman/CEC2009Results.html>> [Diakses 26 November 2017]
- Lucas, S. M., 2009. Ms Pac-Man Competition Software Kit. [program komputer] <<http://cswww.essex.ac.uk/staff/sml/pacman/pac.zip>> [Diakses 2 November 2017]
- Lucas, S. M., 2011. Ms Pac-Man Competition. [online] <<http://cswww.essex.ac.uk/staff/sml/pacman/PacManContest.html>> [Diakses 26 November 2017]
- Microsoft, 1998. Microsoft Revenge of Arcade. [program komputer] <<http://www.old-games.com/download/8896/microsoft-revenge-of-arcade>>
- Midway, 1982. Ms. Pac-Man Parts and Operating Manual. Franklin Park, Illinois 60131. USA.
- Tan, T. G. & Teo, J. (2016, Januari). Application of Artificial Intelligence Techniques in Ms. Pac-Man Game: A Review. Malaysia.
- Thawonmas, R. & Matsumoto, H. (2009, Oktober). Automatic Controller of Ms.

Pac-Man and Its Performance: Winner of the IEEE CEC 2009 Software Agent Ms. Pac-Man Competition. Graduate School of Science and Engineering, Ritsumeikan University, Japan.