دانشگاه صنعتي اميركبير
(پلی تکنیک تهران)
دانشکده ریاضی و علوم کامپیوتر

درس هوش مصنوعی

پروژه دوم

نام نویسنده
ایلیا راوند

استاد درس
دکتر مهدی قطعی

تدریسیار
دکتر بهنام یوسفی

چکیده

مقاله شامل الگوریتم مسیر یابی به وسیله آ استار هندسی است.

کل دیگاه به مسئله بر روی یک دستگاه خودران پیاده سازی شده و به هدف دوری از مسیر طولانی و پیچ های زیاد است.

در این الگوریتم از همان آ استار قبلی استفاده میشود با این تفاوت که رئوس در لیست بسته توست دو تابع p و w فیلتر میشوند و خانه ها حذف میشوند تا بتوان مسیر مطلوب را بدست اورد.

و همینطور برای دوری از پیچ های با درجه زیاد پیچ ها با منحنی ب-اسپلاین مکعبی جایگذین میشود.

بررسی این الگوریتم با ۷ الکوریتم دیگر نشان میدهد تعداد نود ها ۱۰ تا ۴۰ درصد کم شده‌اند تعداد پیچ ها ۲۵ درصد و طول کل مسیر ۲۵.۵ درصد کاهش یافته است.

واژه‌های کلیدی:

جست‌وجوی عمق اول

جست‌وجوی سطح اول

جستجوی اول عمق عمقی تکراری

آ استار

آ استار هندسی

فهرست مطالب **صفحه**

چکیده.....ا
فصل اول: مقدمه ای کوتاه از مقاله.....۱
۱-۱- چکیده مقاله.....۱
۱-۲- طرح مسیله.....۱
۱-۳- مقدمه.....۱
فصل دوم :نحوه مدل کردن تصویر.....۱
۲-۱- شطرنجی سازی محل بندر.....۱
۲-۲- نمایش شبکه ای.....۲
۲-۳-عملیات شناسایی موانع.....۲
فصل سوم : الگوریتم *A ۲
فصل چهارم: نتایج تحلیل پیاده سازی الگوریتم ها ۳

1- مقدمه ای کوتاه از مقاله

۱-۱-چکیده:

موضـــوع در حال بررســـی ما یافتن بهترین مســـیر بین دو نقطه از بندر هســـتش بر پایه الگوریتم Geometric A*. مسایل دیگه که ما پوشش میدهیم در این الگوریتم این است که وسیله نقلیه ما در حال رانندگی به مکان هایی که در آن باید چرخش انجام بدهد میرسد با زاویه مناسبی چرخش هارا انجام بدهد تا باعث رانندگی امن تری شود.

۱-۲-طرح مسئله:

ما در یک بندر ماشین های هوشمندی داریم که باید بین دو نقطه از بندر جابه جا شوند.

چالش هایی که این ماشین با آن رو به رو هستند از این قبیل است:

-یافتن بهترین مسیر(نه فقط به منظور کوتاه ترین)

-موانعی که ممکن است با آن مواجه شود

-مسیر هایی با چرخش های تیز که ممکن است به بار ها آسیب برساند(زاویه چرخش زیاد)

۱-۳-مقدمه:

مسئله برای حل شامل چند بخش است.

۱-مدل کردن بندر برای توانایی پردازش تصویر

۲-بررسی انواع الگوریتم‌های جست‌وجو برای پیدا کردن کوتاه ترین مسیر

برای این بخش ۷ الگوریتم گفته میشود و مشکلات هر یک گفته میشود

که خلاصه‌ای از انها در پایین امده است.

Voronoi diagrams : این الگوریتم تصویر ما را تبدیل میکند به یک سری از چندضلعی های پیوسته میکند. محیط ما با رنگ های مختلف تفاوت بین مانع و مســـیر را مشـــخص میکند و پس از آن میتوان شروع کرد به مسیر یابی. مزیت این الگوریتم این است که تضمین میکند مسیر پیدا شده امن است ولی فضای زیادی اشغال میکند و مسیر بهینه را لزوما پیدا نمیکند.

Visibility Graphs : این الگوریتم اطلاعات هندســی گرفته شــده را تحلیل میکند که موانع را در نظر می گیرند و انها را با چند ضلعی های مناسبی جایگزین می کنند. مزیت این الگوریتم این است که بصری اسـت و کوتاه ترین مسـیر را می توان به راحتی تعیین کرد. اما هنگامی که موقعیت نقطه شـروع و نقطه هدف تغییر می کند، کل نقشه باید بازسازی شود، که انعطاف پذیری ضعیفی را ارائه می دهد.

Grid Structure : در مقایسه با دوتا الگوریتم بالا این الگوریتم کارآمد ترین است چرا که ذخیره سازی و پیاده سازی و مدیریت راحت تری دارد و حتی نیاز به تغییر و بروزرسانی هم ندارد.

۳-بررسی الگوریتم A star

۴-بهینه سازی مسیر برای حرکت وسیله و چالش ها

الگوریتم مورد بررسی ممکن است مسیرهای نامطلوب پیشنهاد دهد پس ما ان را بهینه تر میکنیم

۵-کمینه کردن درجه پیچ ها برا عبور راحت دستگاه

۲-نحوه مدل کردن تصویر:

ابتدا تصویر از ماهواره های فضایی گرفته میشود

۲-۱-شطرنجی سازی محل بندر:

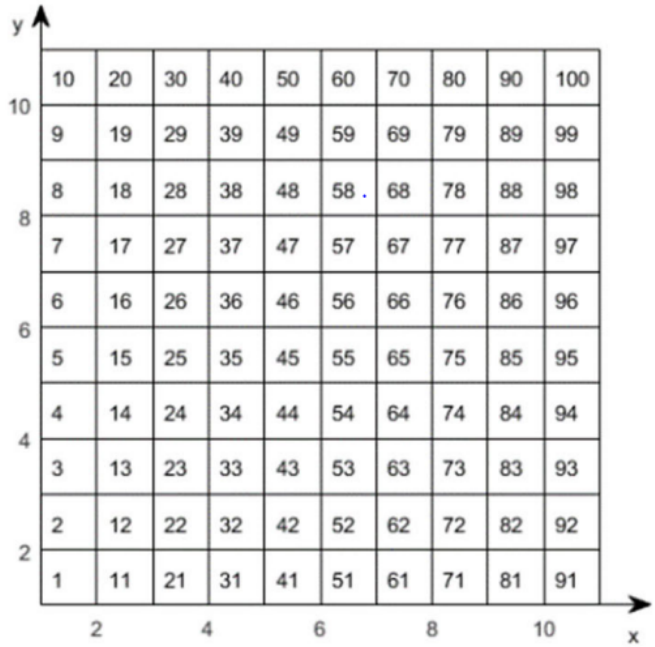
ما نقشه شبکه ای گرفته شده را بررسی میکنیم و به صفحه شطرنجی با واحد لازمه تبدیل میکنیم.

مرحله بعد موانع را در دنیای واقعی پیدا میکنیم و آن را در شــبکه شــطرنجی شــده به رنگ مشــکی در می‌آوریم و مسیر های قابل دسترسی را به رنگ سفید.

۲-۲- نمایش شبکه ای:

پس از تبدیل تصویر داده شده به شبکه شطرنجی مسیله ما تبدیب میشود به یک مسیله بهینه سازی مسیر.

صـفحه شـطرنجی همانطور که گفته شـد به واحد یکسـانی واحدگذاری شـده پس آن را میتوان با یک نقشـه مختصاتی نمایش داد.



۲-۳- عملیات شناسایی موانع:

برای این که خودرو به موانع برخورد نداشته باشد یک فرمول معرفی خواهیم کرد که در طول الگوریتک پیدا کردن مسیر آن باید حفظ شود.

$$di\ (x, y) > rA + rO$$

در فرمول بالا نماد rA یک شعاع امن که ما تعریف میکنیم از خودرو است و rO همان شعاع امن از مانع است. di(x, y) فاصله خودرو از مانع است.

1- الگوریتم *A_هندسی:

این الگوریتم بر پایه الگوریتم *A است که راه های غیر منطقی(cross path and swatooth path) را از آن حذف میکنیم.

در این جا ما دوتا تابع تعریف میکنیم به نام های p(x , y) و w(x , y) که آن ها وظیفه ی فیلتر کردن مسیر های به وجود آمده را دارند.

1-3- الگوریتم *A_هندسی:

الگوریتم ما مراحل زیر را اضافه بر الگوریتم پایه دارد:

۱: پس از قرار گرفتن خانه هدف در لیست بسته با فراخانی کردن تابع های p(x, y) و w(x, y) خانه هایی که دیدن آن ها در مسیر واجب نبوده از لیست حذف خواهند شد.

۲: پس از حذف شدن خانه های اضافی از لیست ما خانه های باقی مانده را بر میگردانیم.

۳: خانه های باقی مانده را به هم وصل میکند و به عنوان بهینه ترین مسیر بر میگرداند.

3-2- مشخص کردن فرمول h(n) :

دو فرمول اصلی برای سوال‌های مسیر یابی در جدول عبارت‌اند از:

1- فاصله منهتن : $h_m(n) = |x_n - x_g| + |y_n - y_g|$

2- فاصله اویلری: $h_E = sqr((x_n - x_g)^2 + (y_n - y_g)^2)$

که در آن (x_n, y_n) مختصات خانه الان ما هست و (x_g, y_g) مختصات نقطه هدف ما است.

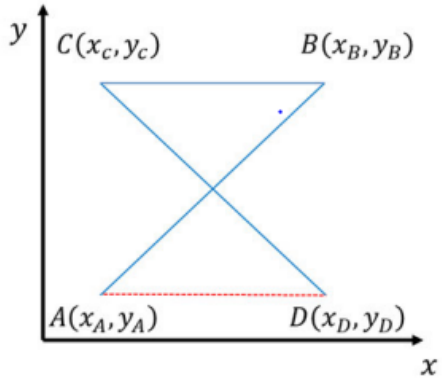
که با توجه به داده ها روش اویلری مسیر کوتاه تری را پیدا میکند پس روش اویلری احتمال بیشتری دارد تا زود تر خانه هدف را پیدا کند. جدول مقایسه:

Distance formula	Number of nodes	Number of turns	Total distance/m
Manhattan	29	1	19.78
Euler	15	0	28

3-3- روش بهبود مسیر های نامنظم در *A_ هندسی:

در روش *A با دو مشکل ممکن بود برخورد کنیم که اسم های آن هارا وضعیت دندانه ای و مسیر های گذرا از مسیر عبور شده گذاشتیم و قرار است دوتابع ارایه دهیم که از این مشکلات جلوگیری کنند.

3-3-1- مشکل مسیر های گذرا از مسیر عبور شده:



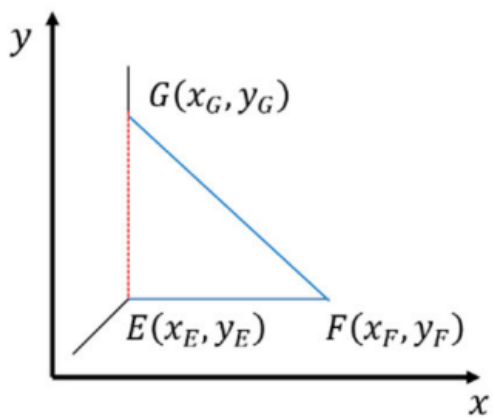
برای این که این مشکل برای ما پیش نیاید ما فضای جستجوی خانه های اطراف را محدود میکنیم.

خانه فعلی و خانه بعدی باید شرایط زیر را داشته باشند:

$$\begin{cases} x_n > x_{n+1} \\ \quad or \\ y_n > y_{n+1} \end{cases}$$

بنابراین ما باید یک تابع داشته باشیم تا این رابطه را برای ما چک بکند که همان p(x, y) ی است که از قبل گفته ایم.

3-3-2- مشکل مسیر دندانه ای:



اینجا هم مانند حالت قبلی باید فضا را محدود بکنیم. اما با چه ضابطه ای؟

$$K_{n,n+1} = \frac{y_{n+1} - y_n}{x_{n+1} - x_n}.$$

ما شیب بین نقطه کنونی و نقطه بعدی را حساب میکنیم اگ شیب منفی بود باید نقطه حذف شود.

۳- الگوریتم A* :

فرض میکنیم یک جدول شبکه‌ای در اختیار داریم و میخواهیم در صورت امکان از نقطه مبدا به مقصد برویم. در این جا از این الگوریتم استفاده میکنیم.

این الگوریتم در هر مرحله خانه با کمترین مقدار در تابع f که خود حاصل جمع دو تابع g و h است را برداشته و g و f مجاوران ان را اپدیت میکند

تعریف g و h به این گونه است:

ng: هزینه رسیدن از مبدا تا نقطه مذکور

nh: تابع هیوریستیک و مقدار حدسی از رسیدن از خانه مذکور به مقصد

۴-نتایج پیاده سازی الگوریتم های پیاده سازی شده:

پیاده سازی جست و جوی عمق اول و جست و جوی سطح اول در پیاده سازی تفوات چندانی نداشتند اما جست و جوی سطح اول در پیدا کردن مسیر سریع تر عمل میکرد و همچنین مسیر کوتاه تری نسبت به الگوریتم دیگر میافت

الگوریتم جســـت و جوی اول عمق عمقی تکراری مســـیر کوتاه تر را میافت ولی زمان بیشـــتری برای محاسبه نیاز داشت

الگوریتم آ استار از تمام الگوریتم های دیگر سریع تر محاسبه میشد اما لزومی بر کوتاه ترین بودن طول مسیر پیدا شده وجود نداشت

منابع و مراجع

There are no sources in the current document.