

پردازش تصویر و کاربرد آن در کنترل روبات مسير ياب

محسن گنج ابراهيم بهروزيان نژاد امين جوادى نسب على شانه سازان زاده
mohsenganj@gmail.com Ebrahim_behrouzian@yahoo.com a_javadi62@yahoo.com mokhtare2000@yahoo.com
دانشگاه آزاد اسلامي واحد شوشتر باشگاه پژوهشگران جوان

چکیده: امروزه روباتها برای واحد بینایی خود از دوربین استفاده می کنند و سپس تصویر گرفته شده را پردازش کرده و سیگنالهای کنترلی را برای روبات به وجود می آورند در این مقاله ما برای کنترل روبات تعقیب خط از علم پردازش تصویر که در برنامه **MATLAB7** می باشد استفاده کرده ایم . برای این روبات هیچ چشم یا سنسوری طراحی نشده است و روبات فقط با استفاده از **web cam** کوچکی که در بالای آن قرار گرفته است از محیط اطراف خود فیلم گرفته و با استفاده از **MATLAB** فیلم گرفته شده را پردازش می کند و سپس از عکس های گرفته شده سیگنال کنترلی را به دست آورده و این سیگنالها را برای کنترل موتورها استفاده می کند.

واژه های کلیدی : port-web cam-Usb – robot – matlab7— Image procesing

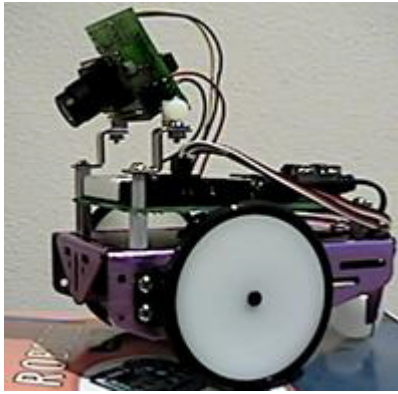
نشان دادن کاربرد کوچکی از علم پردازش تصویر است که می تواند در کنترل روباتهای هوشمند مورد استفاده قرار گیرد. بدون شک یکی از مهمترین مشخصات هوشمندی یک روبات واحد بینایی آن است که امروزه در روباتهای به جای استفاده از سنسورها در واحد بینایی از نرم افزارهایی همچون تشخیص الگو (پردازش تصویر) استفاده می نماید .

۲ – ساختار کلی روبات

روباتی که طراحی شده است شامل سه قسمت زیراست که ما در این پروژه بحث اصلی روی قسمت نرم افزاری که همان

۱- مقدمه

پردازش تصویر گام اول و اساسی در هر سیستم مبتنی بر بینایی ماشین است. پردازش تصویر دانش جدیدی است که سابقه آن به پس از اختراع رایانه های رقمی باز می گردد با این حال این علم نوپا در چند دهه اخیر از هردو جنبه نظری و علمی پیشرفتهای چشمگیری داشته است. سرعت این پیشرفت به اندازه ای بوده است که هم اکنون و پس از این مدت نسبتا کوتاه به راحتی می توان رد پای پردازش تصویر را در بسیاری از علوم مشاهده کرد . هدف ما از نوشتن این مقاله

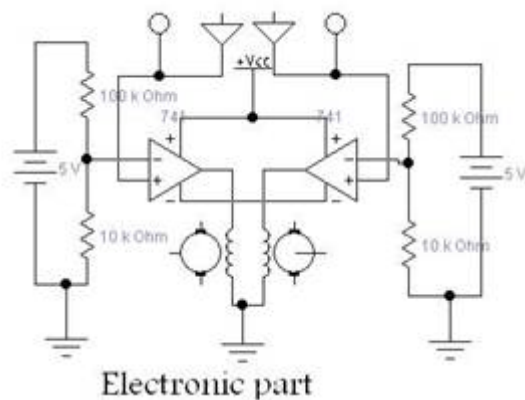


پردازش تصویر و matlab7 است می باشد و قسمتهای دیگر به طور خلاصه گفته می شود.
در یک دید کلی روبات شامل سه قسمت زیر است.

- قسمت مکانیکی
- قسمت الکترونیکی
- قسمت نرم افزاری

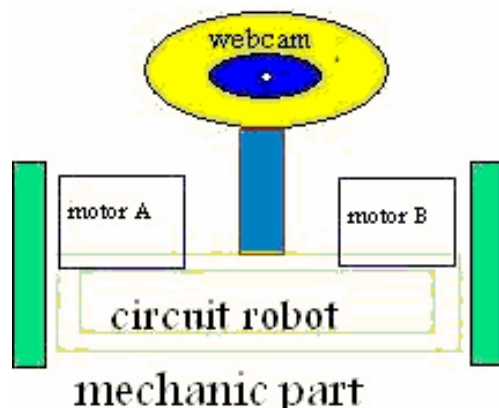
۲-۱- قسمت مکانیکی

قسمت مکانیکی این روبات بسیار ساده بوده ولی دارای ویژگیهای بسیار خوبی است که می تواند به راحتی حرکت کرده و در هر جهتی ازادانه حرکت کند. این طرح که به طرح تانک معروف است فقط دارای دو جرخ است که یکی در طرف راست و یکی در طرف چپ قرار دارد که با استفاده از یک دیفرانسیل ساده نیروی موتورها به چرخها انتقال می یابد و با استفاده از حرکت موتورها به جلو و عقب روبات به راحتی می تواند در هر جهتی که خواست حرکت کند. همانطور که گفته شد این روبات هیچ چشم یا سنسوری ندارد. و برای واحد بینایی آن از یک web cam کوچک استفاده شده است که در بالای آن قرار گرفته است و می تواند صفحه جلوی روبات را ببیند. و تصویر را به کامپیوتر منتقل کند. ساختار کلی قسمت مکانیکی در شکل زیر آمده است.



۲-۳- قسمت کامپیوتری

نرم افزاری که برای اینکار استفاده شده است برنامه MATLAB7 است که دارای ویژگی های جدیدی نسبت به نسخه های قبلی آن است و ویژگی مهمی که ما از آن استفاده کرده ایم این است که MATLAB7 می تواند با پورت USB خود ارتباط برقرار کند در صورتی که در نسخه های قبلی آن این ویژگی امکان نداشت و ما از ویژگی پردازش تصویر آن استفاده کرده ایم. برای این کار لازم است که



تصویر بردار و قابلیت رقمی سازی سیگنال خروجی حسگر می باشد.

مرحله دوم: پس از اینکه تصویر رقمی به دست آمد، مرحله ی بعدی پیش پردازش آن است. وظیفه ی اصلی پیش پردازش، بهبود تصویر به روش هایی است که امکان توفیق سایر پردازشها را نیز افزایش دهد. در این پروژه، پیش پردازش، به طور معمول به روش هایی برای ارتقای تمایز، حذف نویز و جدا سازی آن نواحی که زمینه نشان دهنده ی احتمال وجود اطلاعات لازم است، می پردازد.

مرحله سوم: مرحله ی بعدی به بخش بندی می پردازد. در تعریف وسیع، بخش بندی فرآیندی است که تصویر ورودی را به قسمت ها یا تشکیل دهنده اش تقسیم می کند. خروجی مرحله بخش بندی معمولاً، داده های پیکسلی خام است که یا مرز یک ناحیه یا تمام نقاط درون آن ناحیه را تشکیل می دهند. در هر دو حالت باید داده ها را به شکل مناسب برای پردازش رایانه ای تبدیل نمود. اولین تصمیمی که باید گرفته شود این است که آیا داده ها باید به صورت مرز یا به صورت یک ناحیه کامل نمایش داده شود. نمایش مرزی وقتی مفید است که مشخصات خارجی شکل نظیر گوشه ها یا خمیدگی ها مورد نظر باشد. نمایش ناحیه ای وقتی مفید است که خواص درونی بخش های تصویر نظیر بافت یا استخوان بندی شکل مورد توجه باشد.

۴- برنامه روبات

برنامه ای که برای این روبات نوشته شده است به صورت M

file بوده که با استفاده از MATLAB7

نوشته شده است

این برنامه شامل چند قسمت است

(۱) شناسایی دوربین یا webcam و ارتباط با آن.

(۲) گرفتن عکس توسط webcam و تبدیل آن به تصویر رقمی.

(۳) انجام عملیات و پردازش تصویر بر روی تصویر رقمی.

(۴) تعریف چشم های روبات.

(۵) شناسایی پورت پرینتر توسط MATLAB و ارسال

اطلاعات بر روی آن.

(۶) تحریک مدار درایور

تصویر گرفته شده را وارد کامپیوتر کنیم که این کار با استفاده از یک web cam کوچکی که در بالای روبات قرار دارد صورت می گیرد.

تصویر گرفته شده به صورت زنده از طریق پورت USB کامپیوتر وارد آن می شود و چون MATLAB می تواند USB را شناسایی کند می تواند تصویر گرفته شده را پردازش کند و در نهایت سیگنال کنترلی را برای کنترل روبات بوسیله پورت پرینتر فرستاده و این کار به صورت مداوم و با یک سرعت ثابت انجام می شود. در یک دید کلی می توان کار روبات را اینگونه توصیف کرد که ابتدا روبات با استفاده از دوربین تصویر جلوی خود را می بیند و این تصویر با استفاده پورت USB به کامپیوتر انتقال داده می شود. و سپس تصویر گرفته شده در کامپیوتر به عکس تبدیل شده و در واقع کار اصلی نرم افزار تجزیه و تحلیل عکس هایی است که به صورت مداوم و تقریباً با سرعت ۷ عکس در ثانیه گرفته می شود و سپس در کامپیوتر باید چشمهای فرضی برای آن طراحی کرد. یعنی چشمهای روبات را در نقاط مختلف صفحه در نظر گرفت و از آن نقاط بتوان سیگنالهای ۰ یا ۱ را بدست آورد.

البته باید توجه کنیم که تصویر گرفته شده یک تصویر واقعی است و ما باید این تصویر را با استفاده از فیلتر به یک تصویر سیاه و سفید تبدیل کرد. بعد از اینکار می توان مثلاً رنگ سیاه را ۰ و رنگ سفید را ۱ در نظر گرفت. در واقع کار اصلی این روبات دنبال کردن یک خط در صفحه می باشد که رنگ و اندازه آن متغیر می باشد است. بعد از اینکه فیلم گرفته شده به عکس تبدیل شد چشمهای فرضی را برای آن در نظر گرفت و سپس این سیگنالها را با استفاده از روابطی که برای آن در نظر می گیریم به دو سیگنال کنترلی تبدیل می کنیم و این دو سیگنال را به روبات منتقل می کنیم.

۳- مراحل اساسی پردازش تصویر

مرحله ی اول: این فرایند، تصویر برداری - یعنی به دست آوردن تصویر رقمی است. انجام دادن چنین کاری نیازمند یک حسگر

از دستور getsnapshot استفاده می کنیم. به صورت پایین .

```
I=Getsnapshot(obj)
```

۴-۳- انجام عملیات و پردازش تصویر بر روی تصویر رقمی. تبدیل شکل : عکسی که گرفته شد از نوع شکل RGB است. برای عملیاتی که می خواهیم بر روی شکل انجام دهیم لازم است شکل را به نوع شکل indexed تبدیل کنیم . با استفاده از دستور پایین.

```
I=rgb2ind(I,colorcube(300))
```

یافتن لبه ها : به صورت پایین عمل می شود.

```
I=edge(I,'sobel',(graythresh(I)*.1))
```

حذف نویز: به ترتیب زیر عمل می کنیم.

```
Se90=strel('line',3,90)
```

```
Se0=strel('line',3,0)
```

```
I=imdilate(I,[se90 se0])
```

```
I=imfill(I,'holes')
```

در شکل های زیر مراحل مختلف از گرفتن عکس تا تبدیل آن به سیگنالهای کنترلی نشان داده شده است

original



۴-۱- شناسایی دوربین یا webcam توسط MATLAB7 . برای این کار باید با جعبه ابزار image acquisition کار کرد.

image acquisition toolbox: این جعبه ابزار برای کنترل دوربین یا webcam به کار می رود. برای استفاده از این جعبه ابزار باید مراحل زیر طی شود.

مرحله ۱: نصب وسیله image acquisition ، که این وسیله می تواند webcam یا دوربین دیجیتال باشد.

مرحله 2: به وجود آوردن موضوع ویدئویی ورودی با استفاده از دستور

```
Obj=videoinput('adaptor name',device id,'format')
```

برای تعیین adaptor name از دستور زیر استفاده می شود

Imaqhwinfo

با اجرای این دستور MATLAB adaptor name را به ما می دهد.

مرحله 3: فعال کردن پنجره نمایش دوربین . این قسمت امکان دیدن عکس های گرفته شده توسط webcam را به ما می دهد. و از آن می توان برای تنظیم نور، جهت دوربین و اندازه پنجره نمایش استفاده کرد. و از دستور زیر استفاده می کنیم.

```
Preview('object name')
```

مرحله 4: تنظیم خصوصیات موضوع .

بعد از ایجاد موضوع ورودی ویدئویی و تماشای آن ممکن است بخواهید خصوصیات شکل از جمله روشنایی آن را تغییر دهید. برای این کار از دستور زیر استفاده می شود.

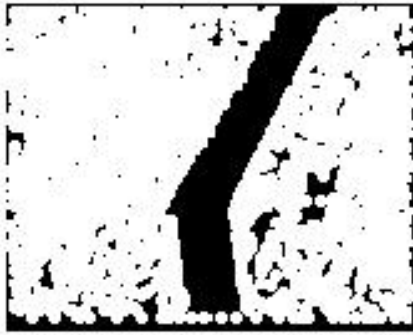
```
Set(obj,'property name',property value)
```

مرحله 5: وقتی که دیگر نیازی به موضوع image acquisition نباشد باید با دستور زیر این موضوع را از فضای کاری MATLAB و حافظه پاک کرد.

```
Delet(obj)
```

۴-۲- گرفتن عکس توسط webcam و تبدیل آن به تصویر رقمی:

figure for imdilate



indexed figure

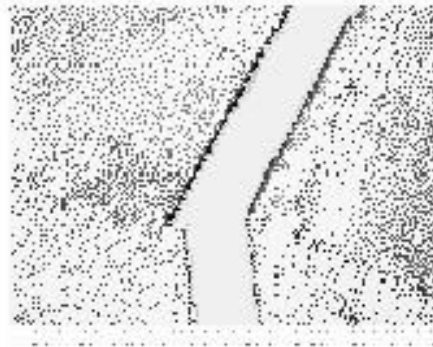


figure for imfill

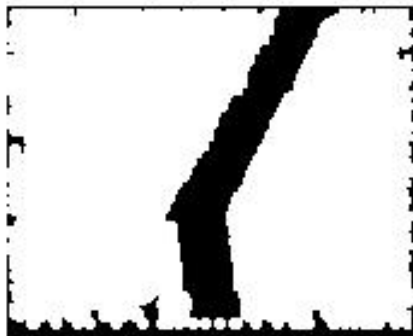


figure edges

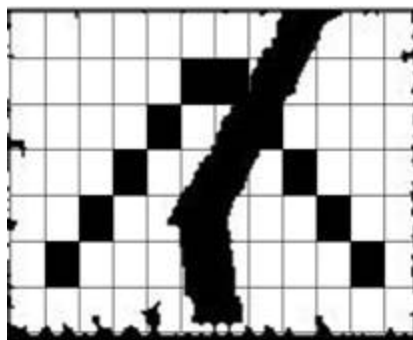
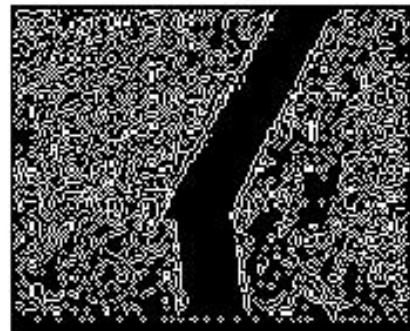


figure for se90

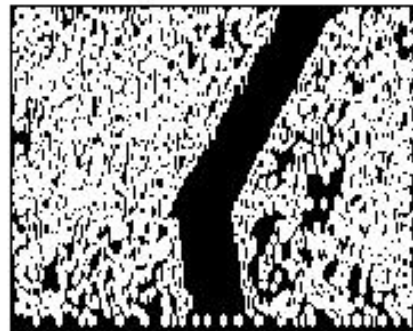
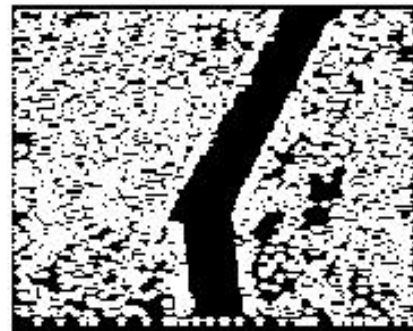
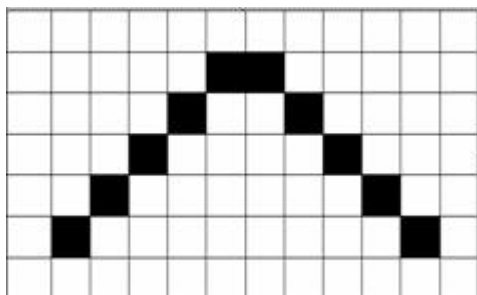


figure for se0



۴-۴- تعریف چشم های روبات

طرز قرار گرفتن چشمهای فرضی در کامپیوتر مانند شکل زیر است



۴-۵- شناسایی پورت پرینتر توسط MATLAB و

ارسال اطلاعات بر روی آن : برای این کار باید با جعبه ابزار data acquisition کار کرد.

data acquisition toolbox : مطلب با استفاده از این جعبه ابزار با پورت پرینتر و کارت صدا ارتباط برقرار می کند. این جعبه ابزار برای ارتباط مطلب با محیط خارج کامپیوتر استفاده می شود.

۴-۶ تحریک مدار درایور

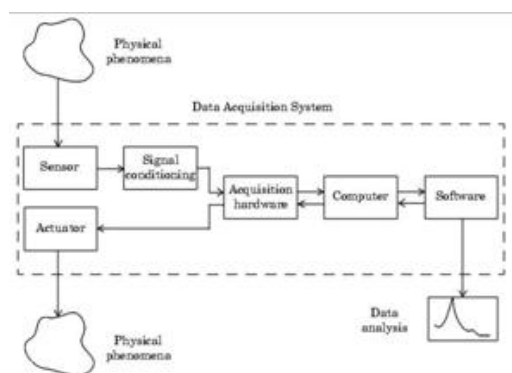
پس از اینکه سیگنالهای کنترلی به وجود آمدند این سیگنالها باید مدار درایور را تحریک کنند . همانطور که در قسمت الکترونیکی دیدید مدار روبات شامل دو ترانزیستور برای هر موتور است که وقتی سیگنال به بیس ترانزیستور می آید آن را تحریک می کند و باعث می شود که رله های مدار قطع و وصل شوند.

۵- اجرای برنامه

```

vid = videoinput('winvideo', 1, 'RGB24_160x120');
preview(vid);
set(vid.source, 'Brightness',40);
for i=1:25
I = getsnapshot(vid);
I = rgb2ind(I,colorcube(300));
PSF = fspecial('gaussian',3,3);
I = imfilter(I,PSF,'symmetric','conv');
BW_s = edge(I,'sobel',(graythresh(I)*.1));
se90 = strel('line',3,90);
se0 = strel('line',3,0);
BWsdil = imdilate(BW_s,[se90 se0]);
w = imfill(BWsdil,'holes');
w = ~w;
imshow(w);
a=w(11:30,73:88);b=w(31:50,57:72);c=w(51:70,41:56);d=w(71:90,25:40);e=w(91:110,9:24);
B=w(31:50,89:104);C=w(51:70,105:120);D=w(71:90,121:136);E=w(91:110,137:152);
fa=mean(mean(a));fb=mean(mean(b));fc=mean(mean(c));fd=mean(mean(d));fe=mean(mean(e));
fB=mean(mean(B));fC=mean(mean(C));fD=mean(mean(D));fE=mean(mean(E));
if fa>.3
fa=1;
else
fa=0;
end;
if fb>.3
fb=1;
else
fb=0;
end;
if fc>.3
fc=1;
else
fc=0;
end;
if fd>.3
fd=1;
else
fd=0;
end;
if fe>.3
fe=1;
else
fe=0;
end;
if fB>.3
fB=1;
else
fB=0;
end;
if fC>.3
fC=1;
else

```



out	STROBE	1	14	AUTOFD	out
	Data 0	2	15	ERROR	in
	Data 1	3	16	INIT	out
	Data 2	4	17	SLCTIN	out
	Data 3	5	18		
	Data 4	6	19		
	Data 5	7	20		
	Data 6	8	21		
	Data 7	9	22	GND	
in	ACK	10	23		
in	BUSY	11	24		
in	PE	12	25		
in	SLCT	13			

حال برای خواندن و نوشتن داده های دیجیتال از پورت پرینتر مراحل زیر را انجام می دهیم.

مرحله ۱: ایجاد یک موضوع دیجیتالی I/O با دستور

```
dio=digitalio('parallel','lpt1')
```

مرحله ۲: اضافه کردن خط و تعریف آن برای خروجی یا ورودی.

```
Addline(dio,2:3,0,'out')
```

مرحله ۳: خواندن یا نوشتن داده ها بر روی پورت. برای مثال

```
data=[1 0]
```

با این دستور عکس گرفته شده به صورت ماتریسی از اعداد در I قرار می گیرد.

است که بهتر است به جای استفاده از سنسور در اینگونه روباتها از دوربین استفاده کرد که امکانات بیشتری دارند.

۷- مراجع

[1] Gonzalez, Rafael C (پردازش تصویر رقمی /

تالیف رافائل سی. گونزالس / ترجمه مرتضی خادمی و

داوود جعفری. _ مشهد: دانشگاه فردوسی مشهد ۱۳۸۲

[2] خود آموز ۶ MATLAB سال انتشار ۱۳۸۰

[3] مهندسی کنترل / کاتسو هیکو اوگاتا / ترجمه محمود

دیانی

HELP MATLAB7[4]

[5] میکرو الکترونیک / عادل صدرا

```
fC=0;
end;
if fD>.3
fD=1;
else
fD=0;
end;
if fE>.3
fE=1;
else
fE=0;
end;
left=fA|fB|fC|fD|fE;
right=fA|fB|fC|fD|fE;
data=[left right]
dio=digitalio('parallel', 'lpt1');
lines=addline(dio,2:3,0, 'out');
putvalue(dio,data);
delete(dio)
end
delete(vid)
```

برای اجرای برنامه کافی است که آن را اجرا کنیم برای اینکار می توان آن را RUN یا کلید F5 را فشار داد پس از اجرای برنامه دو تصویر گرافیکی دیده می شود که یکی از آنها تصویر واقعی از محیط اطراف و دیگری تصویری است که به صورت سیاه یا سفید است.

این یک برنامه ON line است یعنی به صورت مداوم کار می کند برای اینکار باید تصویر گرفته شده را به عکس تبدیل کرد و از هر عکس سیگنال کنترلی بدست آید البته هر چه سرعت عکس گرفتن بیشتر باشد سرعت روبات هم بیشتر خواهد بود برای اینکه بتوان چشمهای فرضی روبات را به دست آورد باید عکس را به ماتریس تبدیل کرد و برای هر چشم مختصاتی را در نظر گرفت در این پروژه برای اینکه احتمال درست بودن سیگنال بیشتر باشد ما مربع کوچکی را که در نظر گرفته ایم اینگونه فرض کرده ایم که اگر مساحت رنگ سیاه آن از حدی بیشتر بود آن را سیاه و اگر از حدی کمتر بود آن راه سفید در نظر بگیرد.

۶- نتیجه گیری

در این مقاله روباتی طراحی شده است که می تواند قابلیت های بیشتری داشته باشد و با استفاده از علم پردازش تصویر می توان دنیای کنونی را راحتتر برای روباتها تشریح کرد و با استفاده از نرم افزارهای جدید می توان برنامه نویسی را راحتتر کرد . و پیشنهادی که در این مقاله داده می شود این