



پروژه درس کامپایلر ترم ۲-۱۴۰۲

**استاد درس:** دکتر سعید پارسا

**عنوان پروژه:** DSL برای کتابخانه OpenCV

**اعضای گروه:**

مهدی اوشنی

محمد مهدی شامخی

علی صمدی فرد

نیوشا یقینی

## مقدمه:

یکی از دلایلی که ما را بر آن داشت که به تحلیل و طراحی زبان خاص دامنه (DSL) در زمینه کتابخانه OpenCV در زبان برنامه نویسی پایتون بپردازیم نیاز روزافزون جامعه به مباحث مرتبط با زمینه های بینایی کامپیوتر و کاربردهای این مبحث در مواردی مانند تشخیص چهره، الگوریتم های دسته بندی تصاویر و ... که اهمیت آن با افزایش میزان استفاده از هوش مصنوعی به شدت افزایش پیدا کرده است.

در DSL طراحی شده ما تنها به بخش های مرتبط با آماده سازی تصاویر در جهت پردازش و استفاده از آنها در بخش های پیچیده تر مانند شبکه های عصبی میپردازیم.

## نیازمندی ها:

در این پروژه به منظور شبیه سازی گرامر مورد نیاز از ابزار `antlr4` و زبان برنامه نویسی پایتون استفاده شده است. کتابخانه های استفاده شده در این پروژه در ذیل آمده:

```
antlr4-python3-runtime==4.13.1
contourpy==1.2.1
cyclor==0.12.1
fonttools==4.53.0
kiwisolver==1.4.5
matplotlib==3.9.0
networkx==3.3
numpy==2.0.0
opencv-python==4.10.0.84
packaging==24.1
pandas==2.2.2
pillow==10.4.0
pydot==2.0.0
pyparsing==3.1.2
python-dateutil==2.9.0.post0
pytz==2024.1
six==1.16.0
tzdata==2024.1
tk==0.1.0
```

# توضیحاتی درباره دستورات پیاده سازی شده و کاربرد آنها:

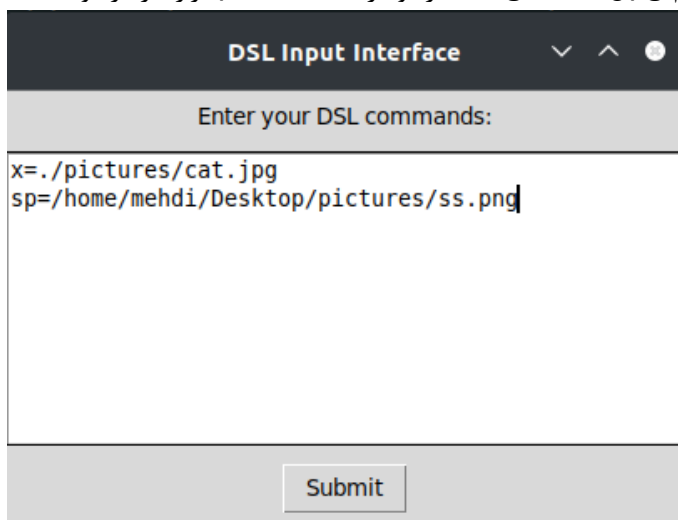
## ۱. دستوری تخصیص تصاویر:

### توضیحات دستور:

این دستور به منظور دریافت یک عکس از ورودی و ایجاد تغییری به منظور استفاده در سایر دستورات مورد استفاده قرار میگیرد.

### نمونه ورودی:

مثلا ما اینجا تصویری به نام `rgbcat.jpg` که در فولدر `pictures` قرار دارد را لود میکنیم.



DSL Input Interface

Enter your DSL commands:

```
x=./pictures/cat.jpg
sp=/home/mehdi/Desktop/pictures/ss.png
```

Submit

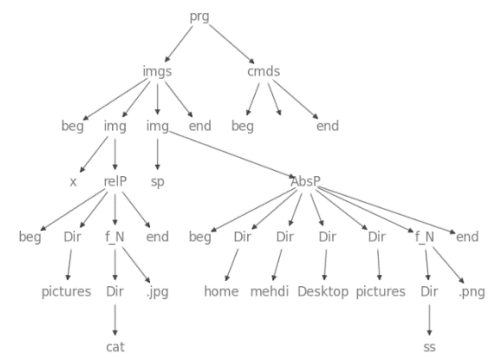
### نکات ورودی:

در این دستور امکان دریافت مسیر عکس به هر دو صورت مطلق و نسبی وجود دارد.

### نمونه خروجی:

این دستور دارای خروجی خاصی به منظور نمایش نمیشد. و بیشتر در سایر دستورات نقش ایفا میکند.

### درخت خلاصه نحوی:



## گرامر:

گرامر این دستور برای خواندن فایل از مسیر های مطلق یا نسبی به صورت زیر است.

```
file: relative_file_path | absolute_file_path;  
relative_file_path: '.' '/' (directory '/')* file_name ;  
absolute_file_path: directory ('/' directory)* '/' file_name;  
file_name: directory file_type;  
file_type: '.png' | '.jpg';
```

```
image_assignments: (image_assignment)+;  
image_assignment: image_var '=' file;|
```

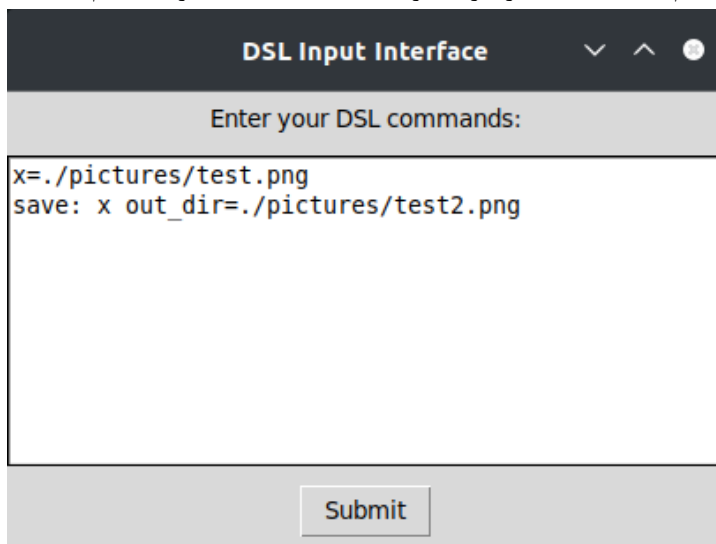
## ۲. دستور ذخیره سازی تصاویر:

### توضیحات دستور:

از این دستور به منظور ذخیره عکس ها پس از اعمال تغییرات در مسیر دلخواه استفاده می شود.

### نمونه ورودی:

مثلا در این جا تصویری به نام test.png را در مسیر ./pictures ذخیره میکنیم.



The image shows a window titled "DSL Input Interface". Inside, there is a text area with the following content:

```
x=./pictures/test.png  
save: x out_dir=./pictures/test2.png
```

Below the text area is a button labeled "Submit".

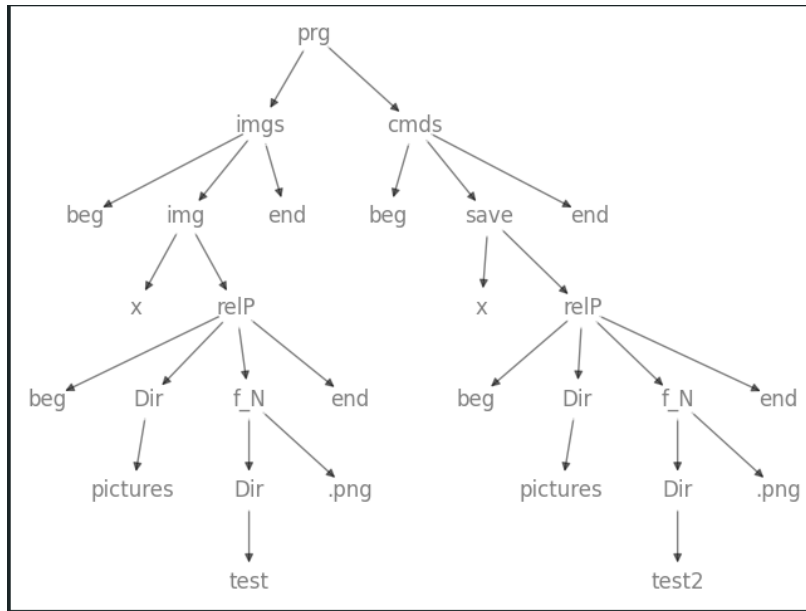
### نکات ورودی:

در این دستور امکان دریافت مسیر عکس به هر دو صورت مطلق و نسبی وجود دارد.

### نمونه خروجی:

این دستور دارای خروجی خاصی به منظور نمایش نمیشد. و بیشتر در سایر دستورات نقش ایفا میکند.

### درخت خلاصه نحوی:



### گرامر:

گرامر این دستور برای ذخیره فایل در مسیر های مطلق یا نسبی به صورت زیر است.

```

save: 'save' ':' image_var 'out_dir' '=' out_dir;
out_dir: file;
  
```

### ۳. دستور نمایش دادن:

#### توضیحات دستور:

از این دستور به منظور نمایش عکس ها پس از اعمال تغییرات در هر مرحله دلخواه استفاده می شود.

#### نمونه ورودی:

در این مثال تصویر test.png را دریافت کرده و در متغیر x ذخیره میکنیم. سپس آن را نمایش میدهیم

DSL Input Interface

Enter your DSL commands:

x=./pictures/test.png  
save: x out\_dir=./pictures/test2.png  
show: x

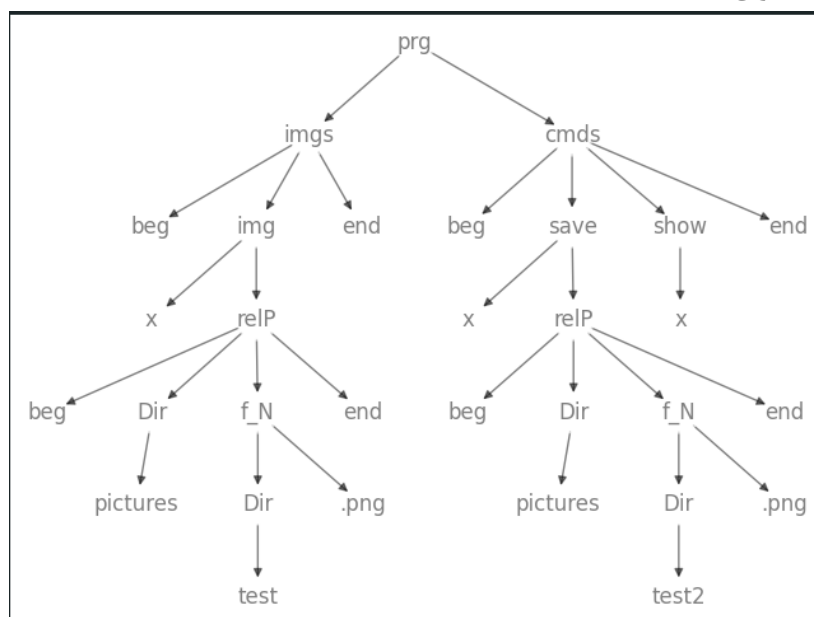
Submit



نمونه خروجی:



## درخت خلاصه نحوی:



گرامر:

```
show: 'show' ':' image_var ;
```

## ۴. دستورات مات کردن

### توضیحات دستور:

از این دستور به منظور مات کردن تصویر با چندین روش استفاده می شود. در صورت انتخاب مات کردن به روش گاوسی میبایست پارامترهای **filtersize** و **std** مقداردهی شود.

در صورت انتخاب مات کردن به روش میانه ای میبایست پارامتر **filtersize** مقداردهی شود.

در صورت انتخاب مات کردن به روش بسته ای (میانگینی) میبایست پارامتر **filtersize** مقداردهی شود.

### نکات ورودی:

در دستوراتی که دارای چندین پارامتر ورودی هستند قابلیت استفاده از جایگشت در پارامترها وجود دارد. یعنی ترتیب ورود پارامترها تأثیری ندارد. به دلیل بزرگ بودن تصویر برای گرامر از آوردن نحوه پیاده سازی آنها اجتناب شده است. در دستور مات گاوسی حتما باید از **filtersize** فرد استفاده کنیم.

## نمونه ورودی گاوسی:

مثلا در دستور زیر ما تصویر `test.png` را به صورت گاوسی مات میکنیم.

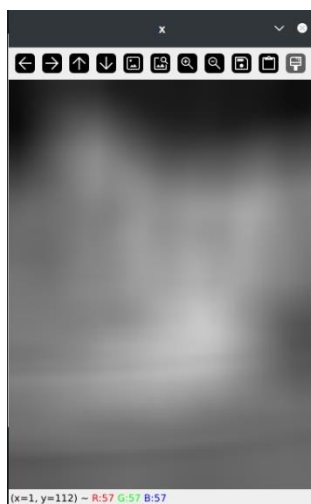
DSL Input Interface

Enter your DSL commands:

```
x=./pictures/test.png
blur: x gaussian -> filtersize = 100.0 , std = 100
show: x
```

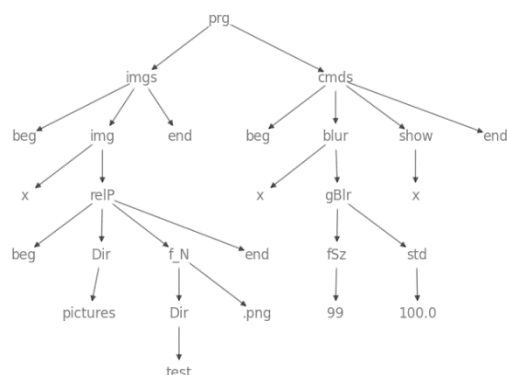
Submit

## نمونه خروجی:





## درخت خلاصه نحوی:



## گرامر:

```

blur_command: 'blur' ':' image_var blur_type ;
blur_type: gaussian_blur | median_blur | box_blur ;
gaussian_blur : 'gaussian' '->' 'filtersize' '=' filter_size ',' 'std' '=' std ;
median_blur : 'median' '->' 'filtersize' '=' filter_size ;
box_blur : ('average' | 'box' ) '->' 'filtersize' '=' filter_size ;
  
```

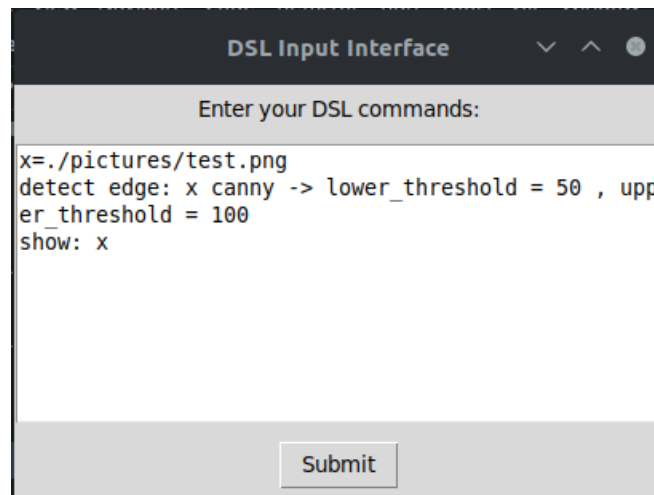
## ۵. دستور شناسایی لبه ها

### توضیحات دستور:

از این دستور به منظور شناسایی لبه های موجود در عکس استفاده می شود. این دستور دارای دو حالت `sobel` و `canny` بوده که در حالت `sobel` میبایست پارامتر `kernel_size` و در حالت `canny` دو پارامتر `lower_threshold` و `upper_threshold` مقداردهی شوند.

### نمونه ورودی:

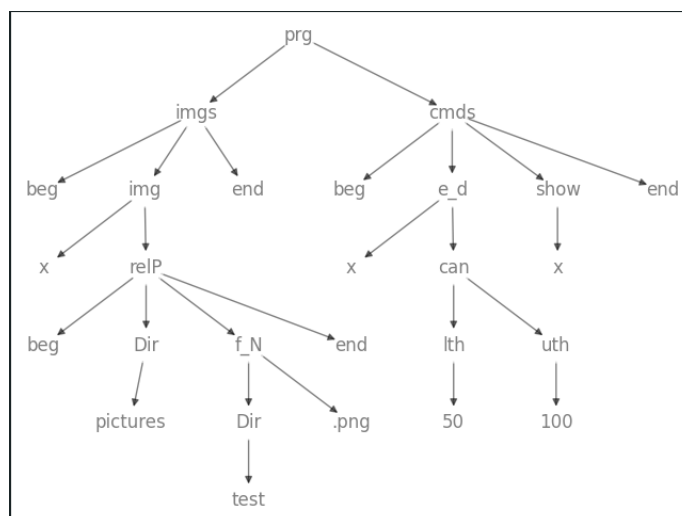
مثلا در دستور زیر ما از دستور `canny` برای یافتن لبه های موجود در عکس `test.png` استفاده کردیم.



نمونه خروجی:



درخت خلاصه نحوی:



گرامر:

```
// Edge Detection
edge_detect_command: 'detect edge' ':' image_var detect_type;
detect_type: sobel_detection | canny_detection ;
sobel_detection: 'sobel' '->' 'kernel_size' '=' kernel_size ;
canny_detection: 'canny' '->' 'lower_threshold' '=' lower_threshold ',' 'upper_threshold' '=' upper_threshold
```

## ۶. دستور کنتراست

توضیحات دستور:

از این دستور به منظور افزایش سطح کنتراست عکس استفاده میشود.

نمونه ورودی:

مثلا در این دستور ما تصویر **test.png** را به عنوان ورودی میدهیم تا کنتراست آن را افزایش داده و سپس نمایش دهد.

DSL Input Interface

Enter your DSL commands:

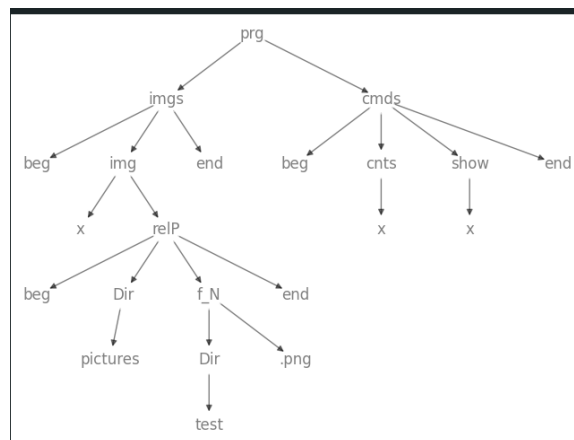
```
x=./pictures/test.png
contrast: x
show: x
```

Submit

نمونه خروجی:



درخت خلاصه نحوی:



گرامر:

```
contrast : 'contrast' ':' image_var;
```

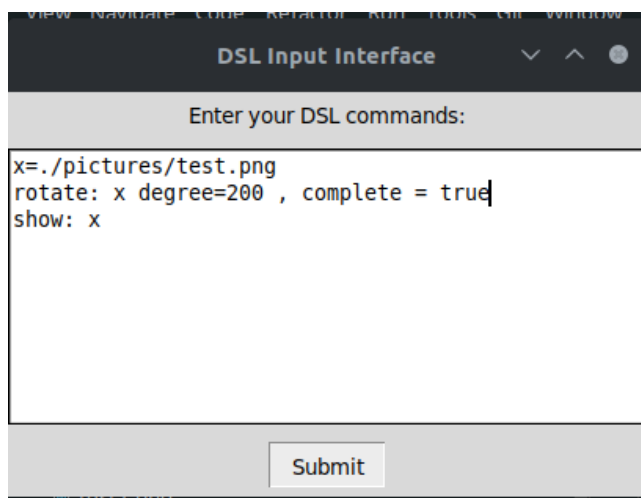
۷. دستور چرخش

توضیحات دستور:

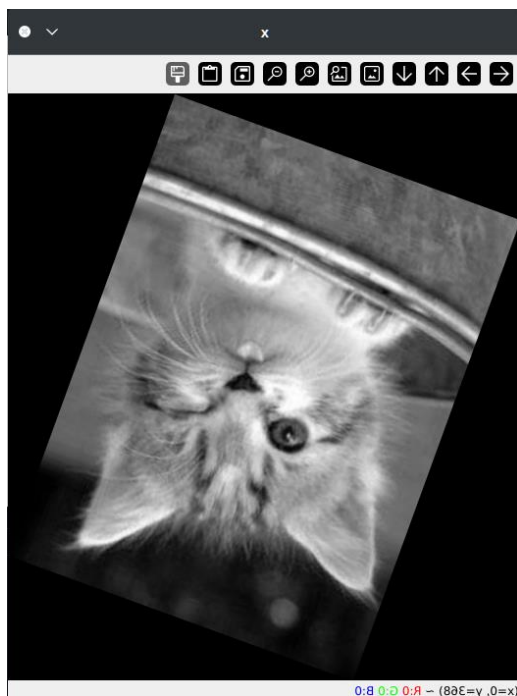
از این دستور برای چرخش تصویر با زاویه مثلثاتی دلخواه استفاده میکنیم

### نمونه ورودی:

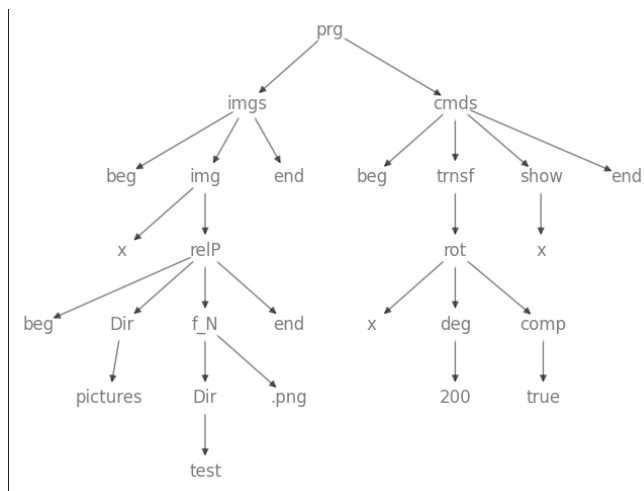
مثلا در دستور زیر ما تصویر `test.png` را به مقدار ۲۰۰ درجه چرخش دادیم و قابلیت پر کردن صفحه رو روشن کرده ایم.



### نمونه خروجی:



درخت خلاصه نحوی:



گرامر:

```
rotation : 'rotate' ':' image_var 'degree' '=' degree ',' 'complete' '=' complete
```

## ۸. دستور ترجمه

توضیح دستور:

این دستور در راستای عمودی و افقی تصویر را جابجا میکند.

نمونه ورودی:

در مثال زیر ما تصویر test.png را به مقدار ۲۰۰ پیکسل در راستای افقی و ۱۰۰ پیکسل در راستای عمود جابجا نمودیم.

DSL Input Interface

Enter your DSL commands:

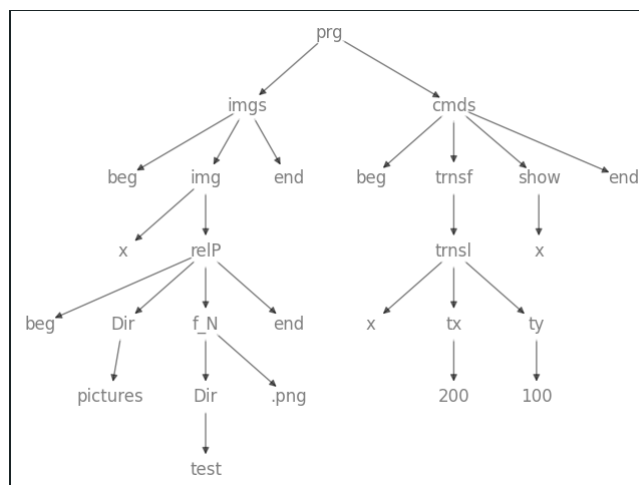
x=./pictures/test.png  
translate: x tx=200 , ty = 100  
show: x

Submit

نمونه خروجی:



درخت خلاصه نحوی:



گرامر:

```
translation : 'translate' ':' image_var 'tx' '=' tx ',' 'ty' '=' ty
```

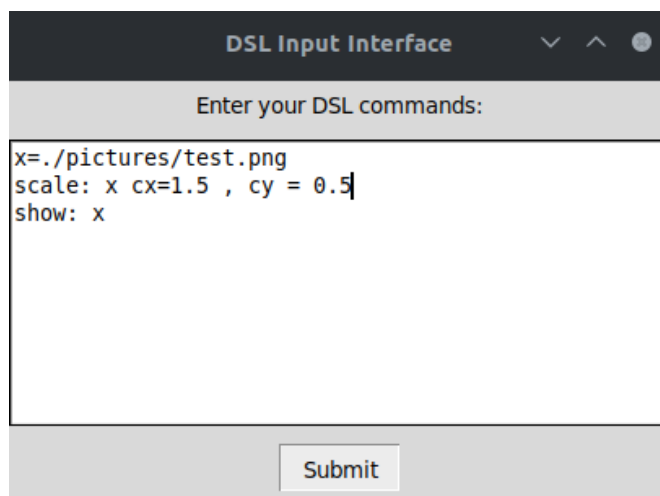
## ۹. دستور مقیاس

توضیح دستور:

از این دستور برای مقیاس دهی تصویر استفاده می شود. این دستور دارای دو پارامتر مقیاس عمودی و افقی است.

### نمونه ورودی:

مثلا در این دستور تصویر `test.png` را تغییر دادیم. مقیاس افقی را ۱.۵ برابر و عمودی را ۰.۵ برابر کردیم.

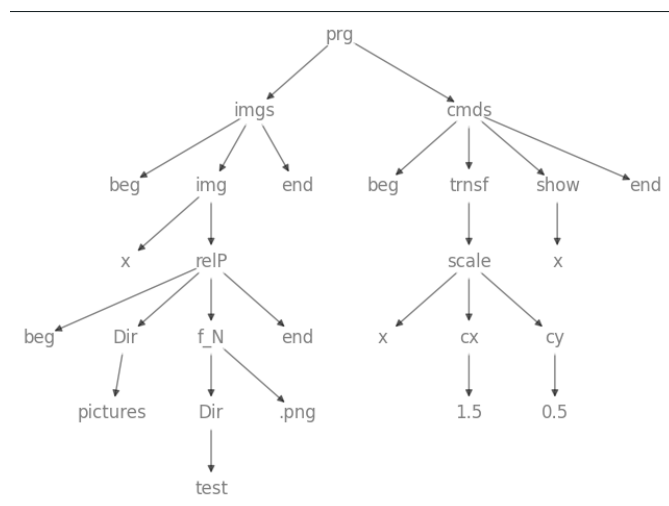


### نمونه خروجی:





درخت خلاصه نحوی:



گرامر:

```
scale: 'scale' ':' image_var 'cx' '=' cx ',' 'cy' '=' cy
```

## ۱۰. دستور برش

توضیح دستور:

از این دستور برای برش عمودی یا افقی با توجه به متغیر `shear_type` و `sh` به عنوان ورودی استفاده میشود.

نمونه ورودی:

مثلا در دستور زیر ما تصویر `test.png` را به صورت عمودی برش میدهم. این کار `shear_type = v` مشخص میکنیم که `v` در آن مخفف `vertical` است.

DSL Input Interface

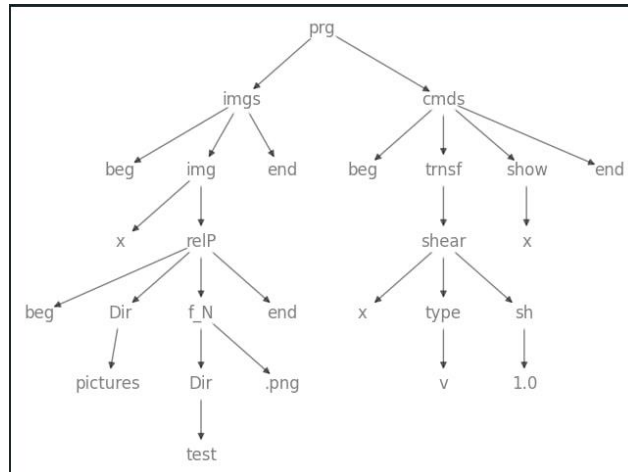
Enter your DSL commands:

```
x=./pictures/test.png
shear: x type= v , sh = 1.0
show: x
```

Submit

نمونه خروجی:

درخت خلاصه نحوی:



گرامر:

```
shear: 'shear' ':' image_var 'type' '=' type_shear ',' 'sh' '=' sh
```

## ۱۱. دستور زوم کردن

### توضیح دستور:

در این دستور ما درصدی به عنوان ورودی می‌دهیم تا بر حسب آن بزرگنمایی یا کوچک نمایی کند. پیکسل وسط عکس به عنوان مرکز زوم کردن استفاده می‌گردد. اگر درصد ورودی ما کوچکتر از ۱۰۰ بود کوچک نمایی و اگر بزرگتر از ۱۰۰ بود بزرگنمایی صورت می‌گیرد.

### نمونه ورودی:

مثلا در کد زیر ما از دستور زوم برای ۱.۵ برابر کردن تصویر استفاده کردیم.

DSL Input Interface

Enter your DSL commands:

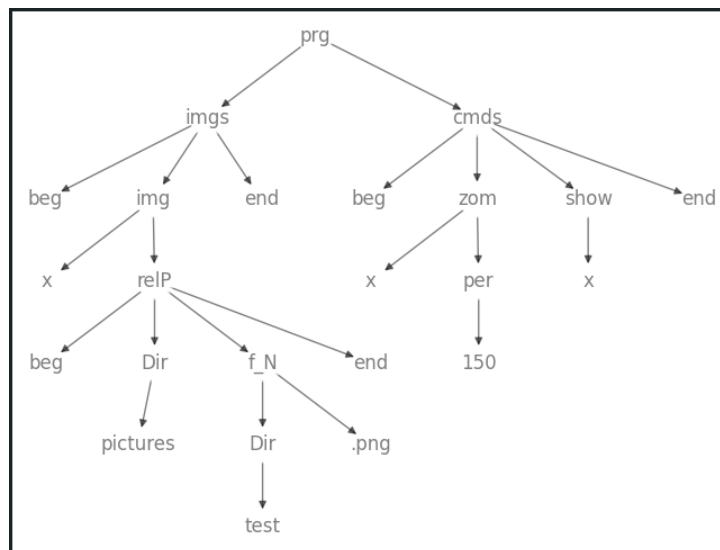
```
x=./pictures/test.png
zoom: x percentage=150
show: x
```

Submit

نمونه خروجی:



درخت اصلاح نحوی:



گرامر:

```
zooming: 'zoom' ':' image_var 'percentage' '=' percentage
```

## ۱۲. دستور تشدید شدت کانال ها

### توضیح دستور:

از این دستور برای تقویت کانال های سه گانه ی عکس ها استفاده می شود. مثلا میتوان شدت رنگ قرمز یا سبز هر پیکسل از عکس را ۲۰ درصد تقویت کرد.

### نمونه ورودی:

مثلا در دستور زیر ما با صفر کردن ضریب سبز و آبی، قرمز تصویر `rgbcat.jpg` را استخراج میکنیم.



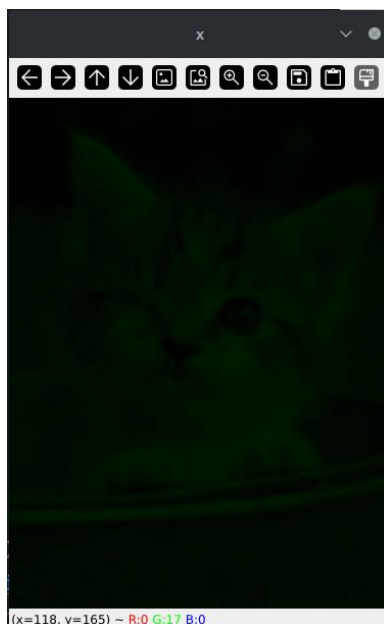
DSL Input Interface

Enter your DSL commands:

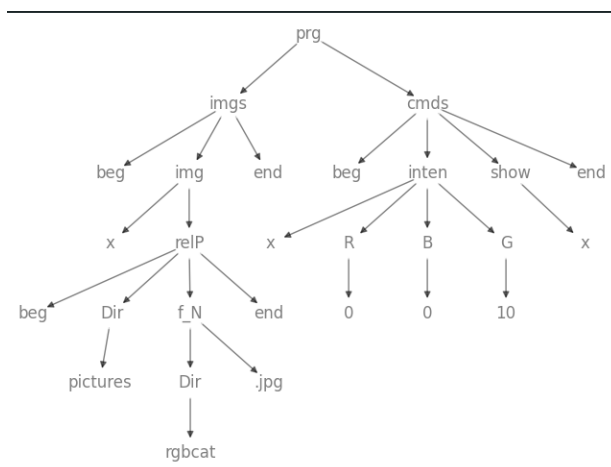
```
x=./pictures/rgbcat.jpg
intensify: x r_ratio=0 , b_ratio=0 , g_ratio=10
show: x
```

Submit

نمونه خروجی:



درخت اصلاح نحوی:



گرامر:

```
intensify : 'intensify' ':' image_var 'r_ratio' '=' r_ratio ',' 'b_ratio' '=' b_ratio ',' 'g_ratio' '=' g_ratio |
'intensify' ':' image_var 'r_ratio' '=' r_ratio ',' 'g_ratio' '=' g_ratio ',' 'b_ratio' '=' b_ratio |
'intensify' ':' image_var 'b_ratio' '=' b_ratio ',' 'r_ratio' '=' r_ratio ',' 'g_ratio' '=' g_ratio |
'intensify' ':' image_var 'b_ratio' '=' b_ratio ',' 'g_ratio' '=' g_ratio ',' 'r_ratio' '=' r_ratio |
'intensify' ':' image_var 'g_ratio' '=' g_ratio ',' 'b_ratio' '=' b_ratio ',' 'r_ratio' '=' r_ratio |
'intensify' ':' image_var 'g_ratio' '=' g_ratio ',' 'r_ratio' '=' r_ratio ',' 'b_ratio' '=' b_ratio ;
```

## ۱۳. دستور سیاه سفید

توضیح دستور:

از این دستور برای سیاه سفید کردن تصاویر رنگی استفاده میکنیم.

## نمونه ورودی:

مثلا در دستور زیر ما تصویر `rgbcat.jpg` را سیاه سفید کردیم.

DSL Input Interface

Enter your DSL commands:

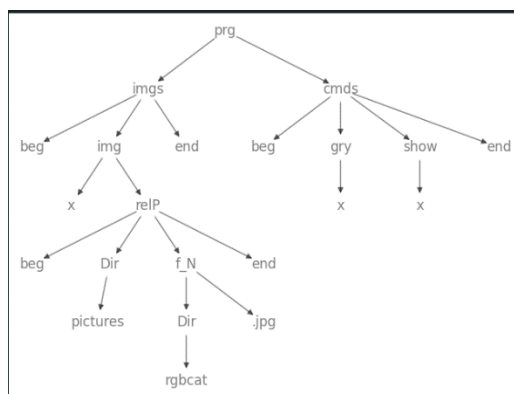
```
x=./pictures/rgbcat.jpg
grayscale: x
show: x
```

Submit

## نمونه خروجی:



## درخت اصلاح نحوی:



گرامر:

```
grayscale : 'grayscale' ':' image_var
```

## ۱۴. دستور حذف نویز:

### توضیح دستور:

از این دستور برای حذف نویزهای تناوبی در تصاویر استفاده می شود. این نحوه از حذف نویز فقط برای تصاویر سیاه سفید بکار میرود و بدین سبب ما ابتدا تصویر را سیاه سفید کرده و سپس برای حذف نویز از روش فوریه استفاده میکنیم. پارامتر ورودی دیگر برای حذف نویز آستانه است.

### نمونه ورودی:

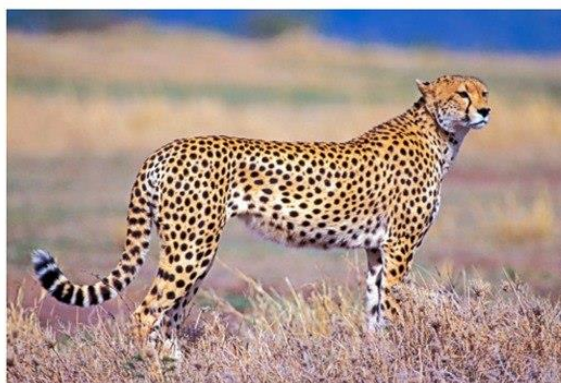
مثلا در این دستور ما تصویر `tilong.jpg` را گرفته و نویز آن را حذف میکنیم.

DSL Input Interface

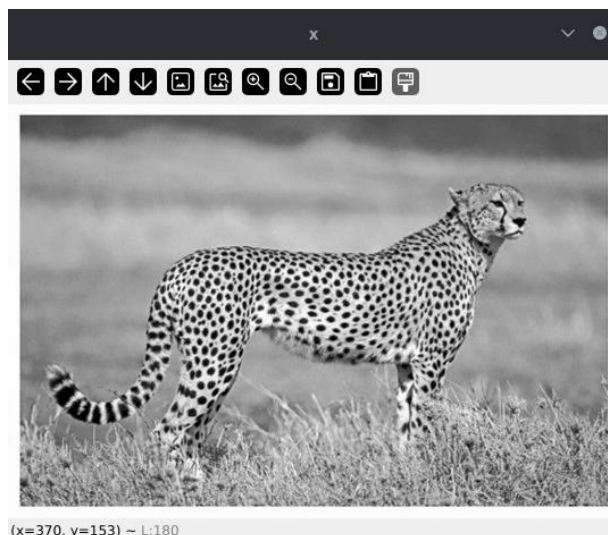
Enter your DSL commands:

```
x=./pictures/tilong.jpg
noise_cancel: x threshold=1000
show: x
```

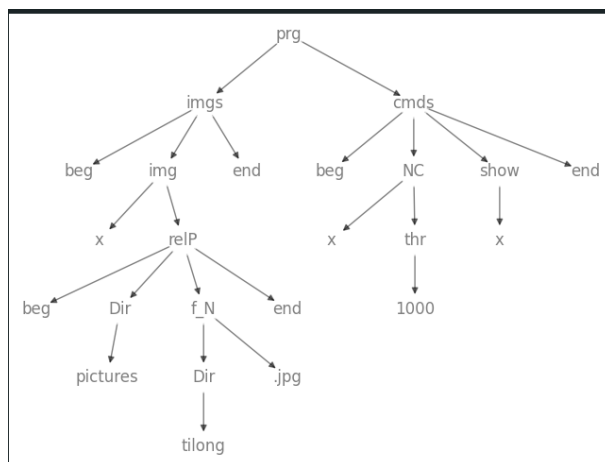
Submit



نمونه خروجی:



درخت اصلاح نحوی:



گرامر:

```
noise_cancelation : 'noise_cancel' ':' image_var 'threshold' '=' threshold;
```

## ۱۵. دستور اضافه کردن نویز

توضیح دستور:

از این دستور برای اضافه کردن نویز نمک فلفل به تصویر دلخواه استفاده میشود. همچنین از پارامتر **amount** برای تنظیم میزان نویز استفاده میکنیم.

نمونه ورودی:

مثلا در این دستور ما روی تصویر **test.png** نویز اضافه میکنیم.



DSL Input Interface

Enter your DSL commands:

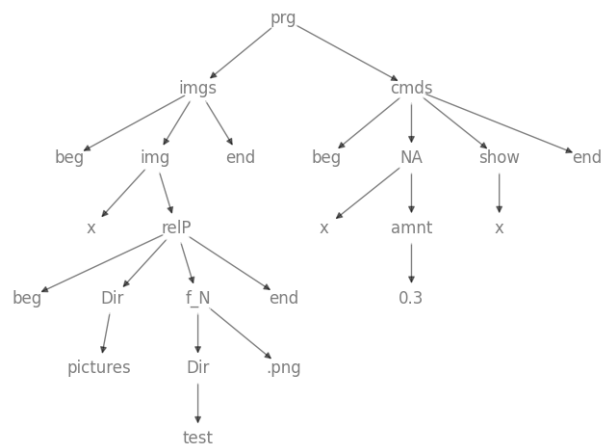
```
x=./pictures/test.png
add_noise: x amount=0.3
show: x
```

Submit

نمونه خروجی:



درخت اصلاح نحوی:



گرامر:

```
add_noise : 'add_noise' ':' image_var 'amount' '=' amount
```

## ۱۶. دستور تشخیص دایره

### توضیح دستور:

این دستور از الگوریتم هاف برای تشخیص دایره در تصویر استفاده میکند. ما به عنوان پارامتر ورودی یک بازه ی قابل قبول برای اندازه شعاع دایره و همچنین کمترین فاصله ی بین مراکز دو دایره را میدهم.

### نمونه ورودی:

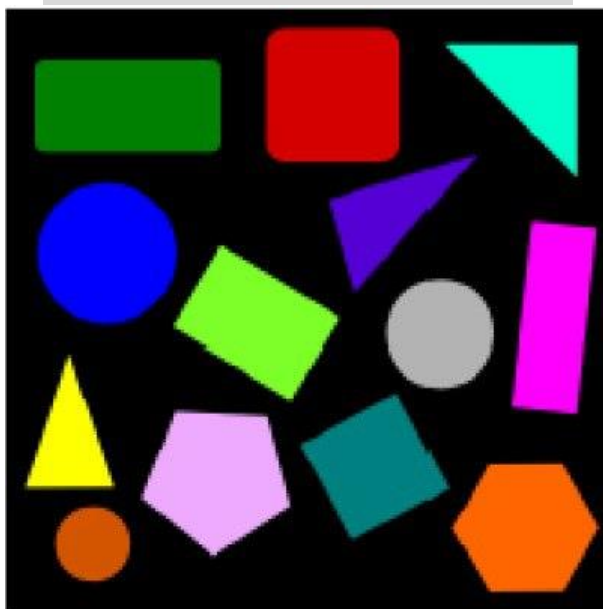
مثلا در نمونه زیر از دستور زیر استفاده میکنیم تا دایره های تصویر `cxx.jpg` را جدا کنیم.

DSL Input Interface

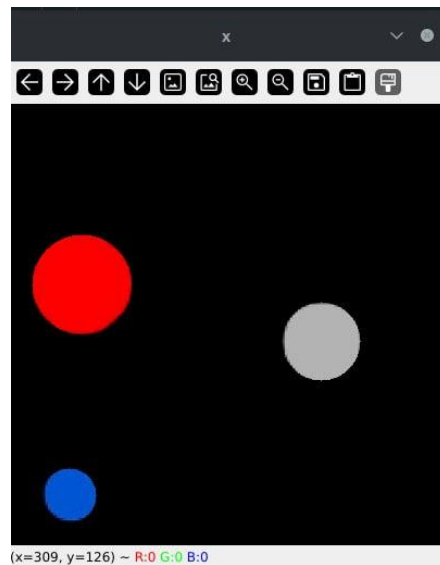
Enter your DSL commands:

```
x=./pictures/cxx.jpg
circle_detect: x min_dist=50 , min_r= 1 , max_r=80
show: x
```

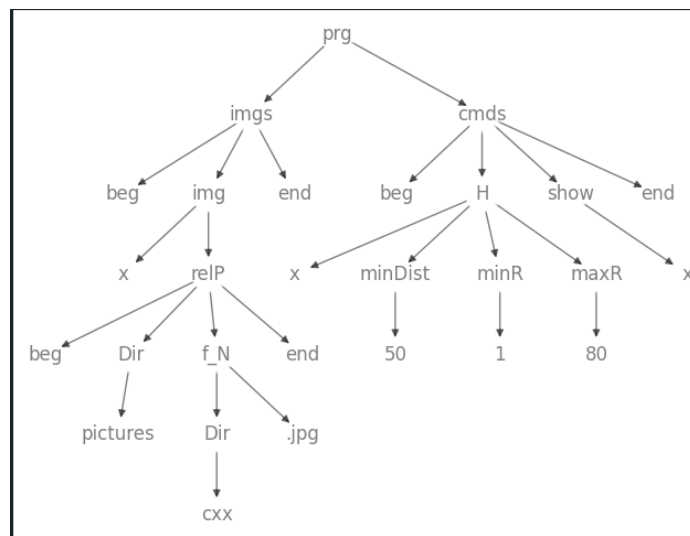
Submit



نمونه خروجی:



درخت اصلاح نحوی:



گرامر:

```

circle_detection: 'circle_detect' ':' image_var 'min_dist' '=' min_dist ',' 'min_r' '=' min_r ',' 'max_r' '=' max_r |
'circle_detect' ':' image_var 'min_dist' '=' min_dist ',' 'max_r' '=' max_r ',' 'min_r' '=' min_r |
'circle_detect' ':' image_var 'max_r' '=' max_r ',' 'min_dist' '=' min_dist ',' 'min_r' '=' min_r |
'circle_detect' ':' image_var 'max_r' '=' max_r ',' 'min_r' '=' min_r ',' 'min_dist' '=' min_dist |
'circle_detect' ':' image_var 'min_r' '=' min_r ',' 'max_r' '=' max_r ',' 'min_dist' '=' min_dist |
'circle_detect' ':' image_var 'min_r' '=' min_r ',' 'min_dist' '=' min_dist ',' 'max_r' '=' max_r;
  
```

ترمینال های تعریف شده در گرامر و تایپ آن ها:

```
directory: FID;
image_var:FID;
filter_size:INTEGER;
std: FLOAT;
kernel_size: INTEGER;
upper_threshold:INTEGER;
lower_threshold: INTEGER;
degree:INTEGER;
complete:BOOL;
tx:INTEGER;
ty:INTEGER;
cx:FLOAT;
cy:FLOAT;
sh:FLOAT;
type_shear:'v' | 'h';
min_dist:INTEGER;
min_r:INTEGER;
max_r:INTEGER;
r_ratio:INTEGER;
g_ratio:INTEGER;
b_ratio:INTEGER;
percentage:INTEGER;
amount:FLOAT;
show_var:BOOL;
save_var:BOOL;
threshold:INTEGER;
```