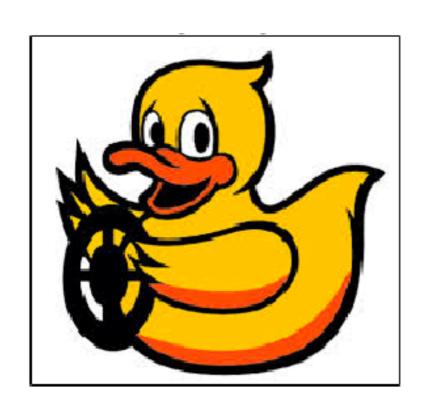
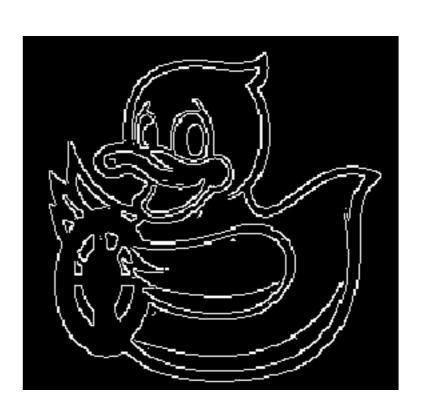
# Lecture 4: Python and OpenCV

呂承龍 Chen-Lung Eric Lu 高熙鈞 KaoCG 李宗儒 ZR,Li

#### **Table of Content**

- Python & Jupyter Notebook
- Line Detector & OpenCV
- Vehicle Detector, Face detector, Duckie Detector





## Python & Jupyter Notebook



If I'm Jupyter

then you are my Sun

being the center of my world till the end of the time

by Jupyter Notebook





- A widely used high-level language
- Emphasize code readability (use whitespace or tab to delimit the code block)
- An interpreted language (no need to compile) 直譯式
- Widely used in data science nowadays

## • python<sup>™</sup>











```
x example.cpp
     example.py
    import rospy
    import numpy as np
    print "This is a python code"
    class test_class(object):
        """docstring for test class"""
        def init (self):
9
10
            self.param_1 = 1
            self.param_2 = []
11
12
            res = self.demontration(5)
13
            print "param_1 = ", self.param_1
14
            print "res = ", res
15
16
17
        def demontration(self, n_loop):
18
            for i in range(0, n_loop):
19
20
                self.param_1 += i
21
22
            return i
23
24
    if __name__=='__main__':
25
26
        obj = test class()
```

```
example.cpp
    #include <stdio.h>
    #include <iostream>
#include <ros/ros.h>
    class test_class
    {
    public:
        test class();
        ~test_class();
10
11
        int16_t param_1;
12
        int16_t param_2[3] = {1, 2, 3};
13
    };
14
    test_class::test_class(){
        param_1 = 1;
16
17
18
        res = this.demonstration(5);
        printf("param_1 = %d\n", param_1);
19
        printf("res = %d\n", res);
20
21
   }
    uint8_t test_class::demonstration(uint8_t n_loop){
24
25
        uint8_t i = 0;
26
27
        for(i=0; i<n_loop; i++){</pre>
28
             param_1 += i;
29
30
31
        return i
32
    }
33
34
    int main(){
        test_class testing();
35
36
        return 0;
37 }
```





```
example.cpp
     example.py
    import rospy
    import numpy as np
    print "This is a python code"
    class test_class(object):
        """docstring for test class"""
        def init (self):
 8
9
10
            self.param_1 = 1
            self.param_2 = []
11
12
            res = self.demontration(5)
13
            print "param_1 = ", self.param_1
14
15
            print "res = ", res
16
        def demontration(self, n_loop):
17
18
            for i in range(0, n_loop):
19
20
                self.param_1 += i
21
22
            return i
23
24
    if __name_ == ' __main__':
25
26
        obj = test class()
```

import/include library

```
example.cpp
1 #include <stdio.h>
    #include <iostream>
#include <ros/ros.h>
    class test_class
    {
    public:
         test class();
        ~test_class();
10
11
        int16_t param_1;
12
        int16_t param_2[3] = {1, 2, 3};
13
    };
14
    test_class::test_class(){
        param_1 = 1;
16
17
18
         res = this.demonstration(5);
        printf("param_1 = %d\n", param_1);
19
        printf("res = %d\n", res);
20
21
    }
    uint8_t test_class::demonstration(uint8_t n_loop){
24
25
        uint8_t i = 0;
26
27
        for(i=0; i<n_loop; i++){</pre>
28
             param_1 += i;
29
30
31
         return i
32
    }
33
34
    int main(){
        test_class testing();
35
36
        return 0;
37 }
```





```
example.cpp
     example.py
    import rospy
    import numpy as np
    print "This is a python code"
    class test_class(object):
        """docstring for test class"""
        def init (self):
8
9
10
            self.param_1 = 1
            self.param_2 = []
11
12
            res = self.demontration(5)
13
            print "param_1 = ", self.param_1
14
            print "res = ", res
15
16
        def demontration(self, n_loop):
17
18
            for i in range(0, n_loop):
19
                self.param_1 += i
20
21
22
            return i
23
24
    if __name_ == '__main__':
25
26
        obj = test_class()
```

🚄 import/include library 🚄

```
example.cpp
    #include <stdio.h>
    #include <iostream>
#include <ros/ros.h>
    class test_class
    {
    public:
         test class();
        ~test_class();
10
11
        int16_t param_1;
12
        int16_t param_2[3] = {1, 2, 3};
13
    };
14
    test_class::test_class(){
16
        param_1 = 1;
17
18
         res = this.demonstration(5);
        printf("param_1 = %d\n", param_1);
19
        printf("res = %d\n", res);
20
21
    }
    uint8_t test_class::demonstration(uint8_t n_loop){
24
25
        uint8_t i = 0;
26
27
        for(i=0; i<n_loop; i++){</pre>
28
             param_1 += i;
29
30
31
         return i
32
    }
33
34
    int main(){
        test_class testing();
        return 0;
```

main function





```
example.cpp
                                                    import/include library 🕳
                     example.cpp
     example.py
                                                                                        #include <stdio.h>
                                                                                        #include <iostream>
#include <ros/ros.h>
    import rospy
    import numpy as np
                                                             class
    print "This is a python code"
                                                                                        class test_class
                                                                                        {
                                                                                        public:
    class test_class(object):
                                                                                             test class();
        """docstring for test class"""
                                                                                             ~test_class();
        def init (self):
8
                                                                                    10
9
                                                                                    11
                                                                                             int16_t param_1;
10
             self.param_1 = 1
                                                                                    12
             self.param_2 = []
11
                                                                                    13
                                                                                        };
12
                                                                                    14
             res = self.demontration(5)
13
                                                                                        test_class::test_class(){
             print "param_1 = ", self.param_1
14
                                                                                    16
                                                                                             param_1 = 1;
             print "res = ", res
15
                                                                                    17
16
                                                                                    18
        def demontration(self, n_loop):
17
                                                                                    19
18
                                                                                    20
                                                                                    21
                                                                                        }
             for i in range(0, n_loop):
19
                 self.param_1 += i
20
21
                                                                                    24
22
             return i
                                                                                    25
                                                                                             uint8_t i = 0;
23
                                                                                    26
24
                                                                                    27
    if __name_ == ' __main__':
25
                                                                                    28
                                                                                                 param_1 += i;
26
        obj = test_class()
                                                                                    29
                                                                                    30
                                                                                    31
                                                                                             return i
                                                                                    32
                                                                                        }
                                                                                    33
                                                                                    34
                                                                                        int main(){
                                                                                             test_class testing();
```

main function

```
int16_t param_2[3] = {1, 2, 3};
    res = this.demonstration(5);
    printf("param_1 = %d\n", param_1);
    printf("res = %d\n", res);
uint8_t test_class::demonstration(uint8_t n_loop){
    for(i=0; i<n_loop; i++){</pre>
    return 0;
```





```
example.cpp
                                                   import/include library 🕳
                     example.cpp
     example.py
                                                                                      #include <stdio.h>
                                                                                      #include <iostream>
    import rospy
    import numpy as np
                                                                                      #include <ros/ros.h>
                                                           class
    print "This is a python code"
                                                                                      class test_class
                                                                                      {
                                                                                      public:
    class test_class(object):
                                                                                          test class();
        """docstring for test class"""
                                                                                          ~test_class();
        def init (self):
8
9
                                                                                          int16_t param_1;
10
             self.param_1 = 1
            self.param_2 = []
11
                                                                                  13
                                                                                      };
                                                        class declare
12
                                                                                  14
            res = self.demontration(5)
13
                                                                                      test_class::test_class(){
             print "param_1 = ", self.param_1
14
                                                                                  16
                                                                                          param_1 = 1;
            print "res = ", res
15
                                                                                  17
16
                                                                                  18
        def demontration(self, n_loop):
17
                                                                                  19
18
                                                                                  20
            for i in range(0, n_loop):
                                                                                  21
                                                                                      }
19
                 self.param_1 += i
20
21
                                                                                  24
22
            return i
                                                                                  25
                                                                                          uint8_t i = 0;
23
                                                                                  26
24
                                                                                  27
    if __name_ == ' __main__':
25
                                                                                  28
                                                                                              param_1 += i;
        obj = test class()
26
                                                                                  29
                                                                                  30
                                                                                  31
                                                                                          return i
                                                                                  32
                                                                                      }
                                                                                  33
                                                                                  34
```

main function

```
int16_t param_2[3] = {1, 2, 3};
    res = this.demonstration(5);
    printf("param_1 = %d\n", param_1);
    printf("res = %d\n", res);
uint8_t test_class::demonstration(uint8_t n_loop){
    for(i=0; i<n_loop; i++){</pre>
int main(){
    test_class testing();
    return 0;
```





```
example.cpp
                                                   import/include library 🕳
                     example.cpp
     example.py
                                                                                      #include <stdio.h>
                                                                                      #include <iostream>
    import rospy
    import numpy as np
                                                                                      #include <ros/ros.h>
                                                           class
    print "This is a python code"
                                                                                      class test_class
                                                                                      public:
    class test_class(object):
                                                                                          test_class();
        """docstring for test class"""
                                                                                          ~test_class();
        def init (self):
8
9
                                                                                          int16_t param_1;
10
             self.param_1 = 1
                                                                                          int16_t param_2[3] = {1, 2, 3};
            self.param_2 = []
11
                                                        class declare
12
                                                                                  14
            res = self.demontration(5)
13
                                                                                      test_class::test_class(){
             print "param_1 = ", self.param_1
14
                                                                                  16
                                                                                          param_1 = 1;
            print "res = ", res
15
                                                                                  17
16
                                                                                          res = this.demonstration(5);
        def demontration(self, n_loop):
                                                                                          printf("param_1 = %d\n", param_1);
17
                                                                                          printf("res = %d\n", res);
18
            for i in range(0, n_loop):
                                                                                  21
                                                                                      }
19
                                                          class define
                 self.param_1 += i
20
                                                                                      uint8_t test_class::demonstration(uint8_t n_loop){
21
                                                                                  24
22
            return i
                                                                                          uint8_t i = 0;
23
                                                                                  26
24
                                                                                  27
                                                                                          for(i=0; i<n_loop; i++){</pre>
    if __name_ == ' __main__':
25
                                                                                              param_1 += i;
                                                                                  28
        obj = test_class()
26
                                                                                  29
                                                                                  30
                                                                                  31
                                                                                          return i
                                                                                  32
                                                                                  33
                                                                                  34
                                                                                      int main(){
                                                                                          test_class testing();
                                                                                          return 0;
                                                      main function
```

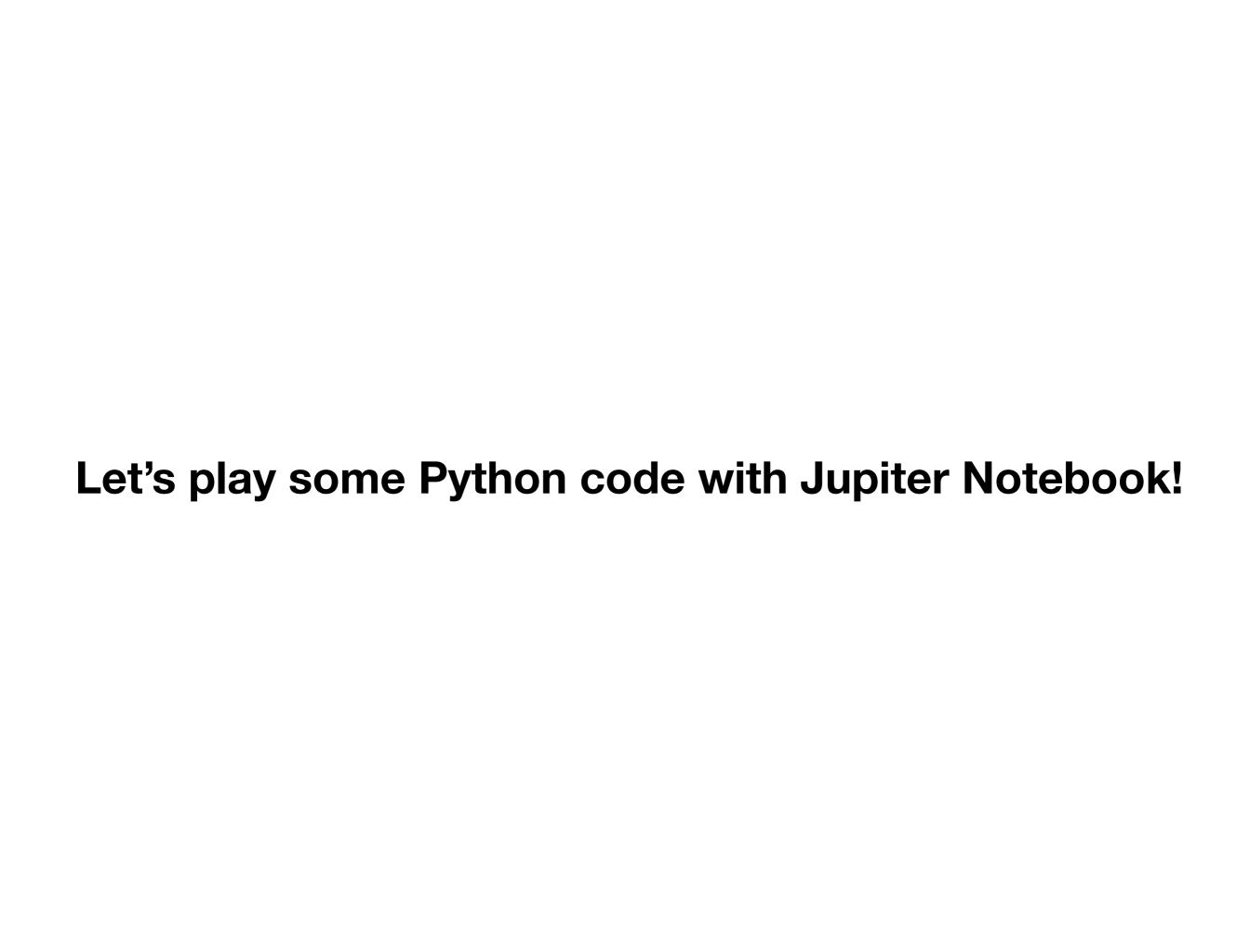




```
example.cpp
                                                   import/include library 🕳
                     example.cpp
     example.py
                                                                                      #include <stdio.h>
                                                                                      #include <iostream>
    import rospy
    import numpy as np
                                                                                      #include <ros/ros.h>
                                                           class
    print "This is a python code"
                                                                                      class test_class
                                                                                      public:
    class test_class(object):
                                                                                          test class();
        """docstring for test class"""
                                                                                          ~test_class();
        def init (self):
8
9
                                                                                          int16_t param_1;
10
             self.param_1 = 1
                                                                                          int16_t param_2[3] = {1, 2, 3};
            self.param_2 = []
11
                                                                                      };
                                                        class declare
12
                                                                                  14
            res = self.demontration(5)
13
                                                                                      test_class::test_class(){
             print "param_1 = ", self.param_1
14
                                                                                  16
                                                                                          param_1 = 1;
            print "res = ", res
15
                                                                                  17
16
                                                                                  18
                                                                                          res = this.demonstration(5);
        def demontration(self, n_loop):
                                                                                          printf("param_1 = %d\n", param_1);
17
                                                                                          printf("res = %d\n", res);
18
            for i in range(0, n_loop):
                                                                                  21
                                                                                      }
19
                                                          class define
                 self.param_1 += i
20
                                                                                      uint8_t test_class::demonstration(uint8_t n_loop){
21
                                                                                  24
22
            return i
                                                                                          uint8_t i = 0;
23
                                                                                  26
24
                                                                                  27
                                                                                          for(i=0; i<n_loop; i++){</pre>
    if __name_ == ' __main__':
25
                                                                                              param_1 += i;
                                                                                  28
        obj = test class()
26
                                                                                  29
                                                                                  30
                                                                                  31
                                                                                          return i
                                                                                  32
                                                                                  33
                                                                                  34
                                                                                      int main(){
                                                                                          test_class testing();
                                                                                          return 0;
                                                      main function
```

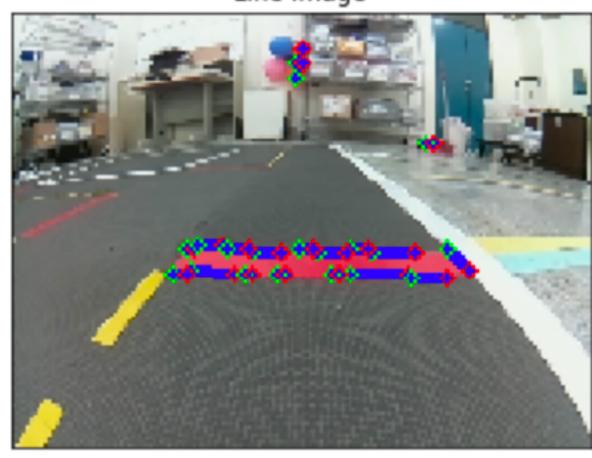


- A web application to create and share document that contain live code.
- Show the code output directly
- Only interpreted code (like python)



## Line Detector & OpenCV

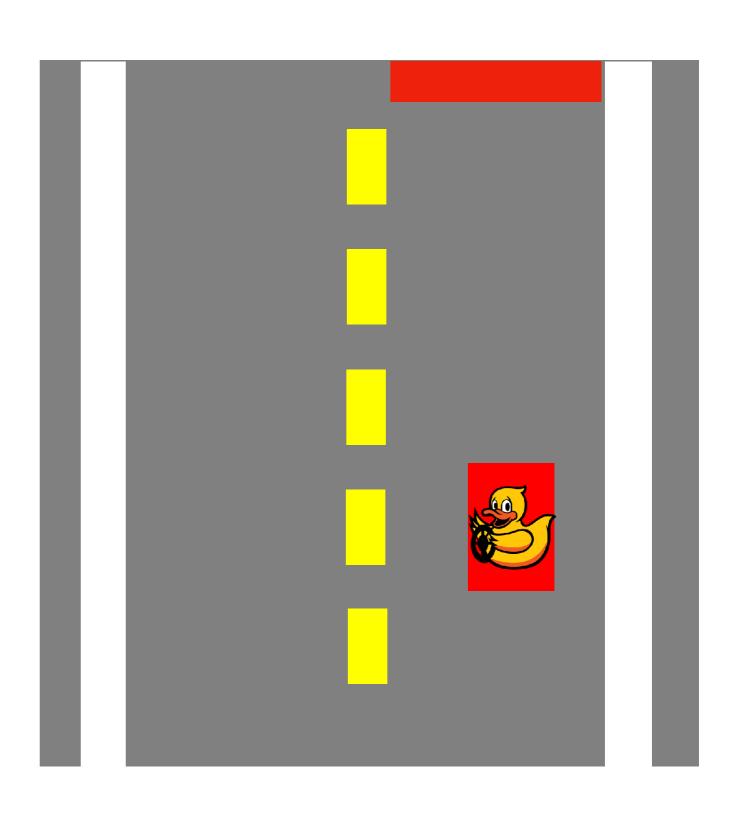
Line Image





## Q: What does Duckiebot detect so that it know its position and angle w.r.t to the road?

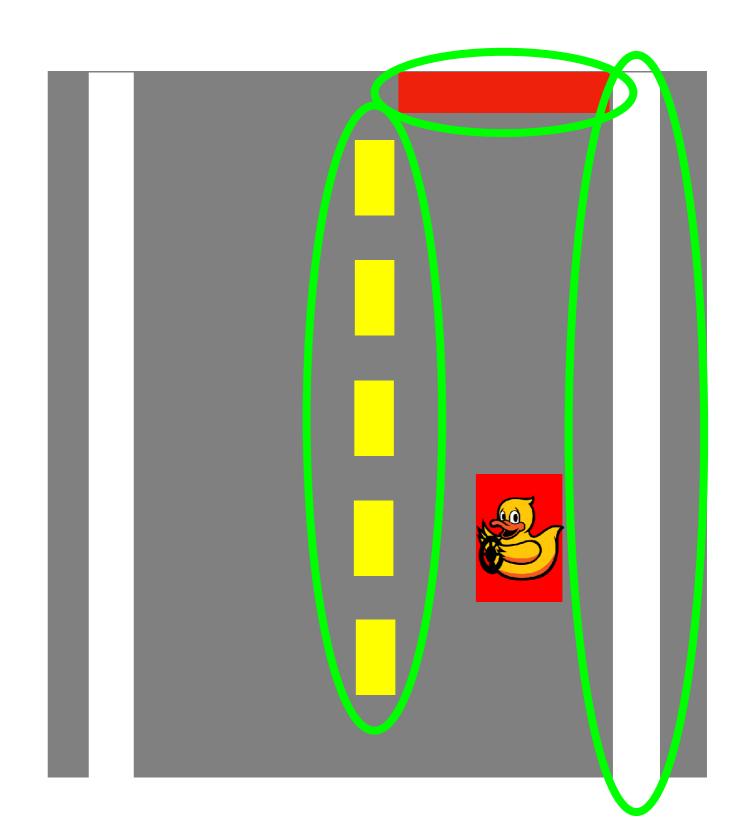




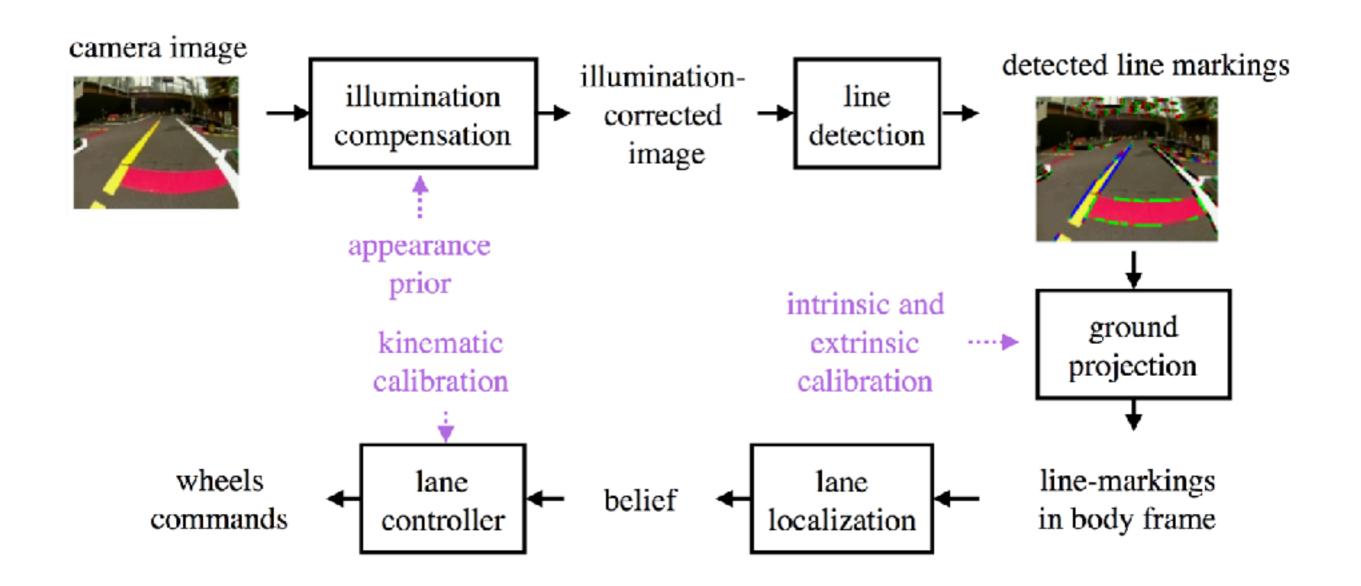
Q: What does Duckiebot detect so that it know its position and angle w.r.t to the road?

## A: Lines!

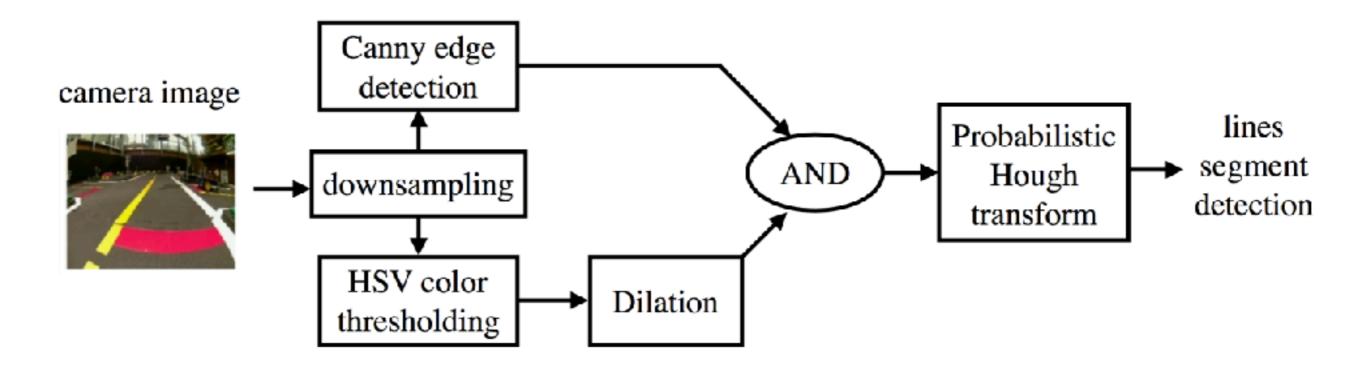


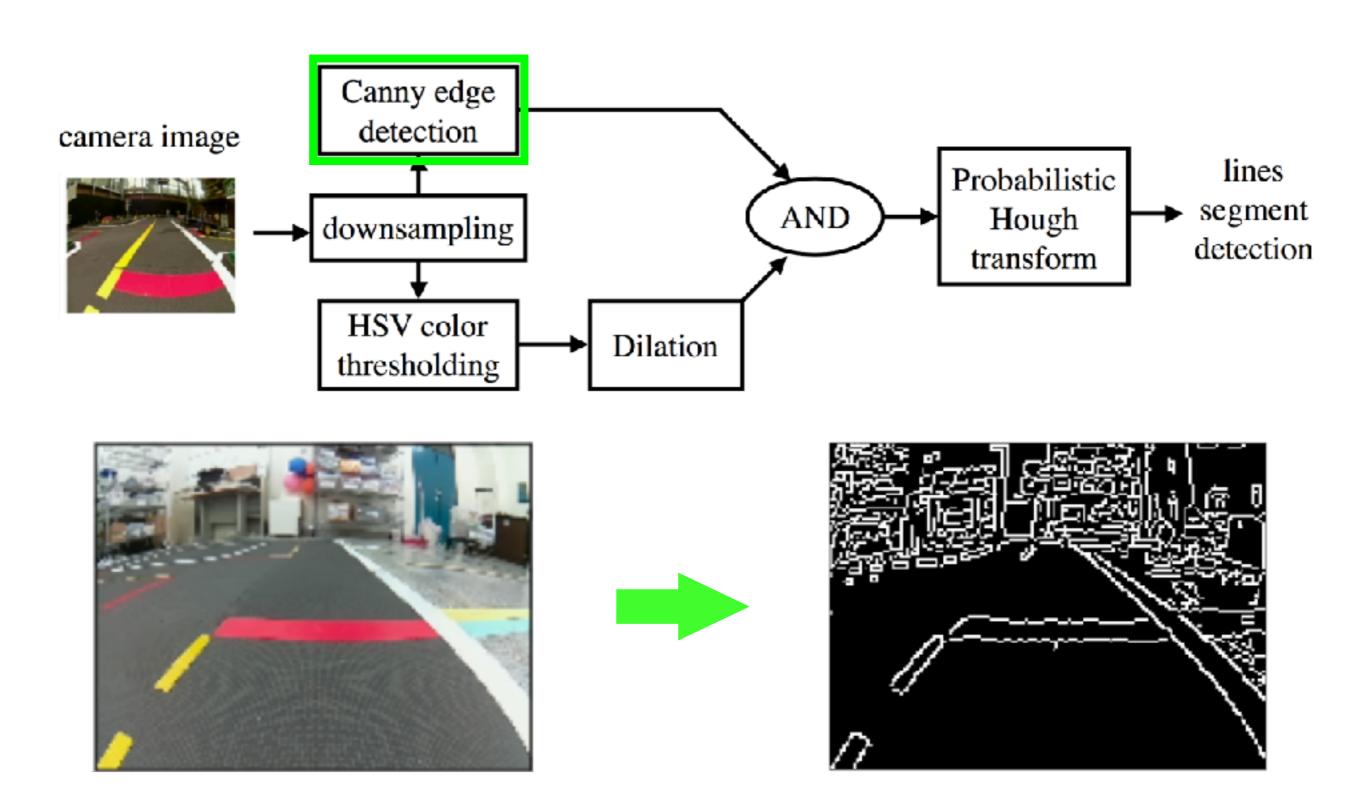


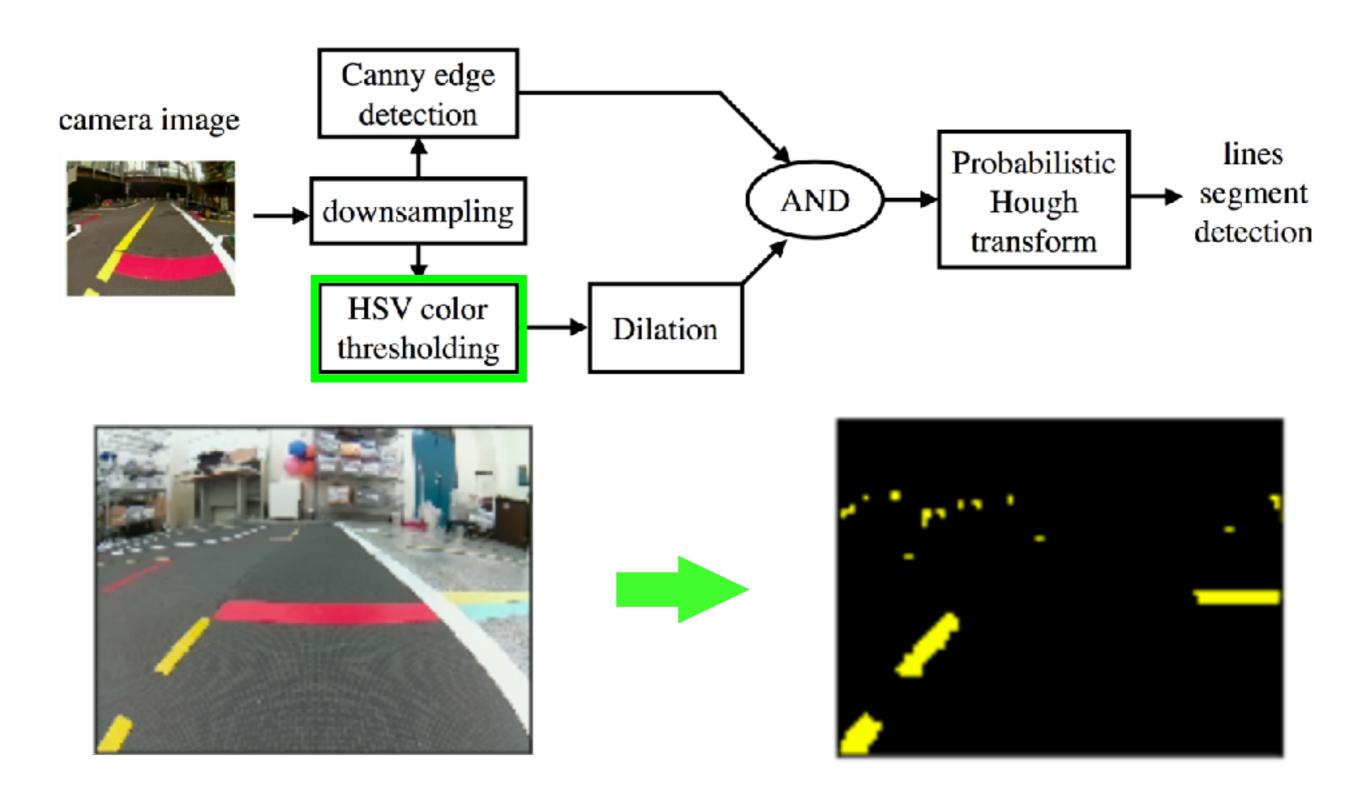
#### Pipeline of Lane Following

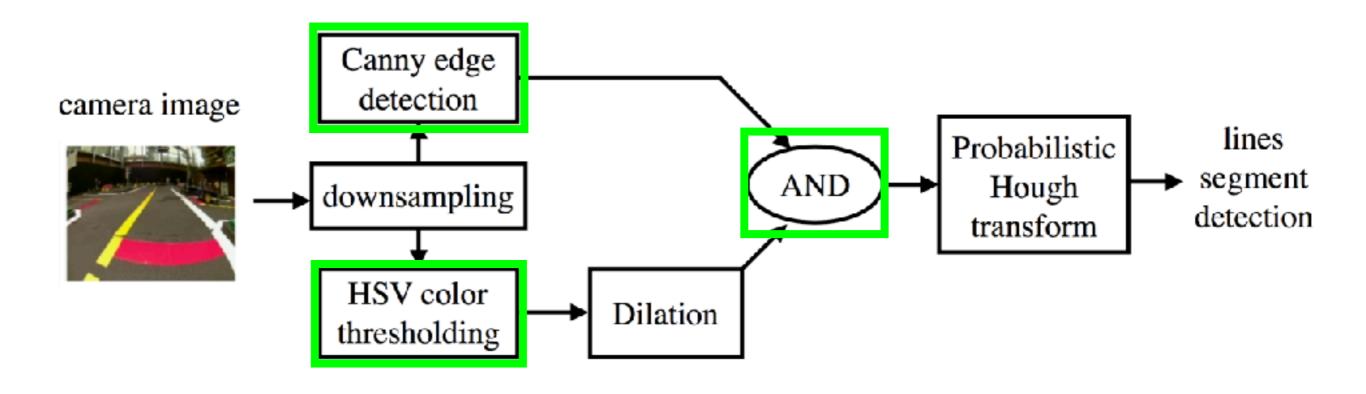


#### Pipeline of Lane Following camera image detected line markings illuminationillumination line corrected compensation detection image appearance prior intrinsic and ground kinematic extrinsic projection calibration calibration wheels line-markings lane lane belief controller localization in body frame commands

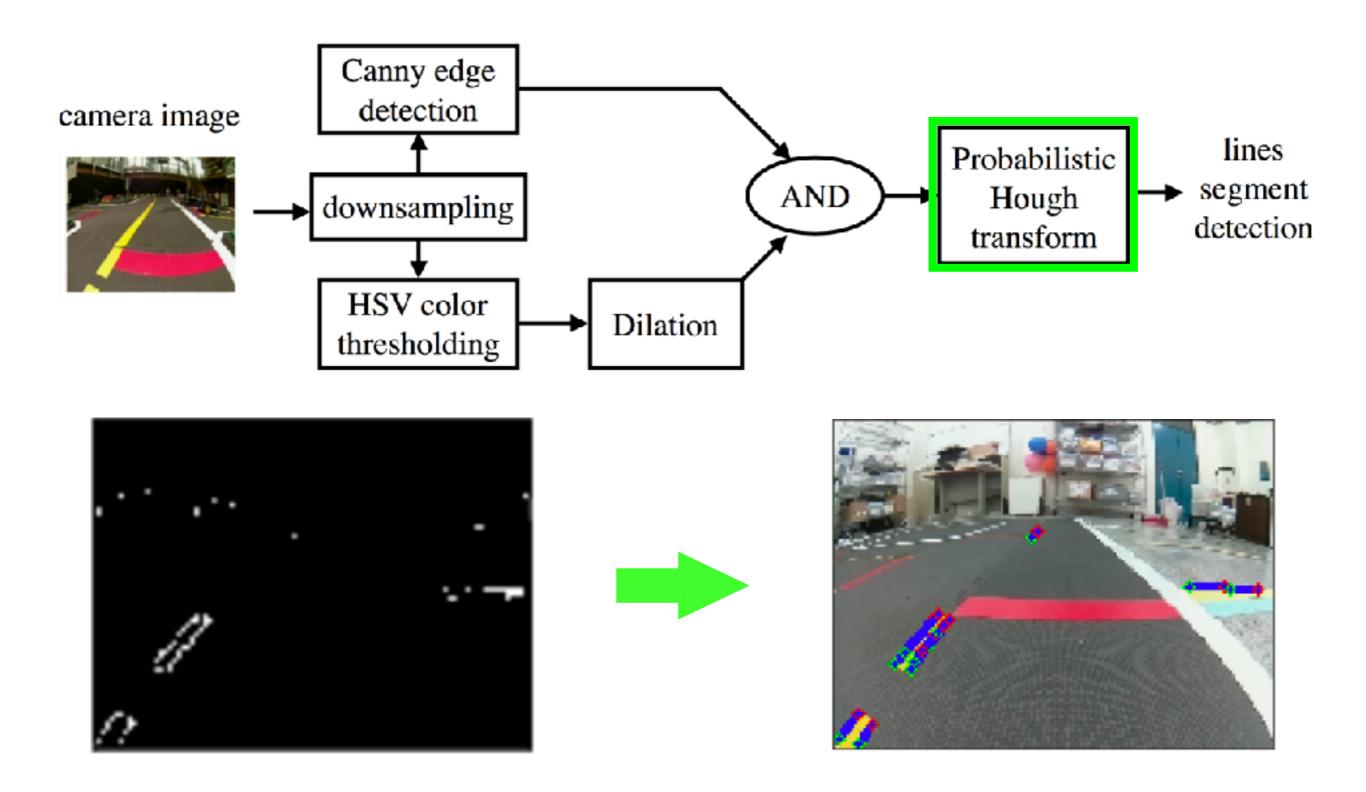








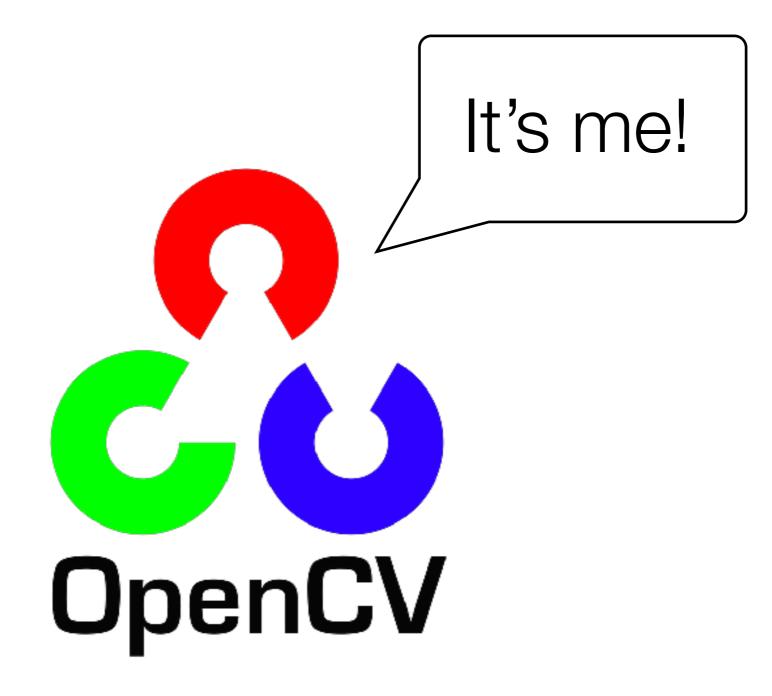




#### How can we perform those magics?



## by OpenCV

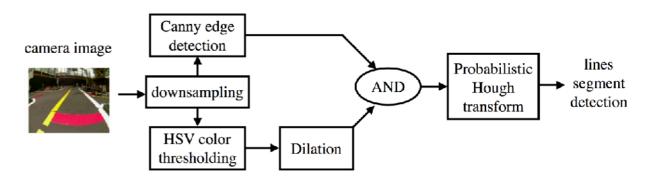




- OpenCV (Open Source Computer Vision Library)
- Open source computer vision and machine learning software library
- Provide a common infrastructure for computer vision (machine learning) applications

Canny edge detector

cv2.Canny(img, minVal, maxVal)



Color space transform

result = cv2.cvtColor(img, cv2.COLOR\_BGR2HSV)

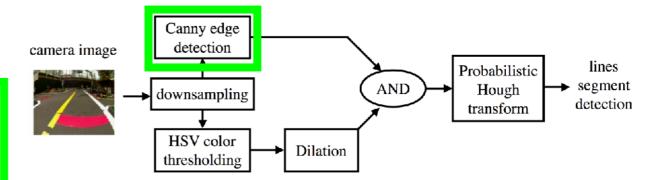
Thresholding

result = cv2.inRange(img, minVal, maxVal)

Probabilistic Hough Transform

Canny edge detector

cv2.Canny(img, minVal, maxVal)



Color space transform

result = cv2.cvtColor(img, cv2.COLOR\_BGR2HSV)

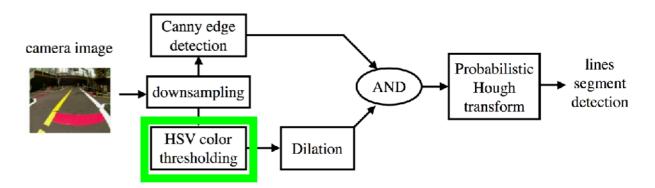
Thresholding

result = cv2.inRange(img, minVal, maxVal)

Probabilistic Hough Transform

Canny edge detector

cv2.Canny(img, minVal, maxVal)



Color space transform

result = cv2.cvtColor(img, cv2.COLOR\_BGR2HSV)

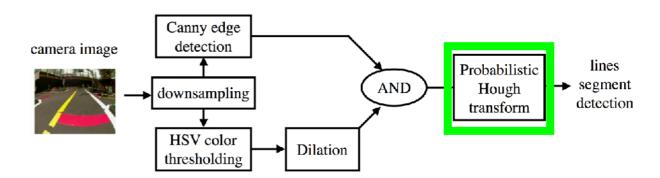
Thresholding

result = cv2.inRange(img, minVal, maxVal)

Probabilistic Hough Transform

Canny edge detector

cv2.Canny(img, minVal, maxVal)



Color space transform

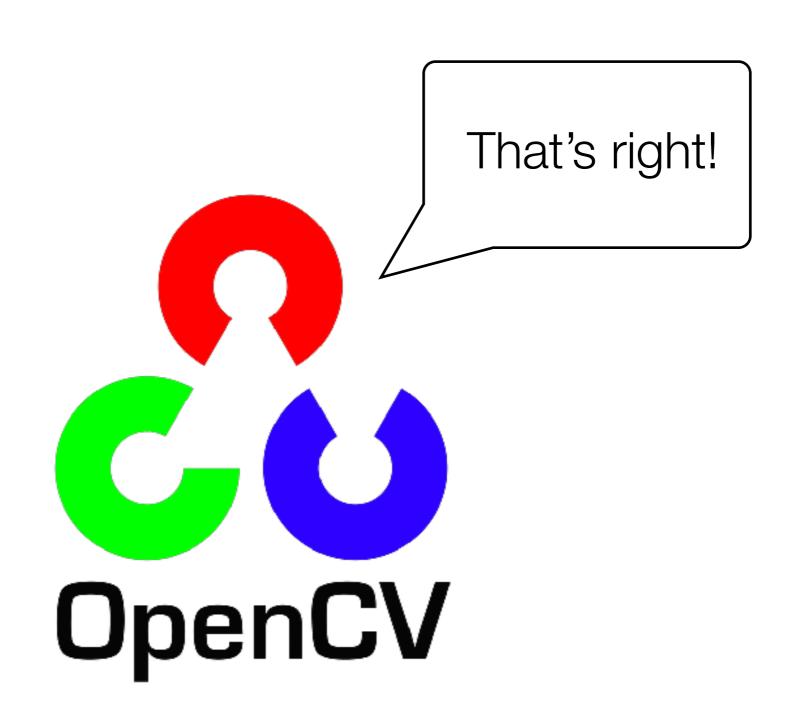
result = cv2.cvtColor(img, cv2.COLOR\_BGR2HSV)

Thresholding

result = cv2.inRange(img, minVal, maxVal)

Probabilistic Hough Transform

#### In conclusion, OpenCV helps us a lot in development



#### Last but not least....

Let's go through some background knowledge of

Canny edge detector

**HSV** color thresholding

How do we detect edges?

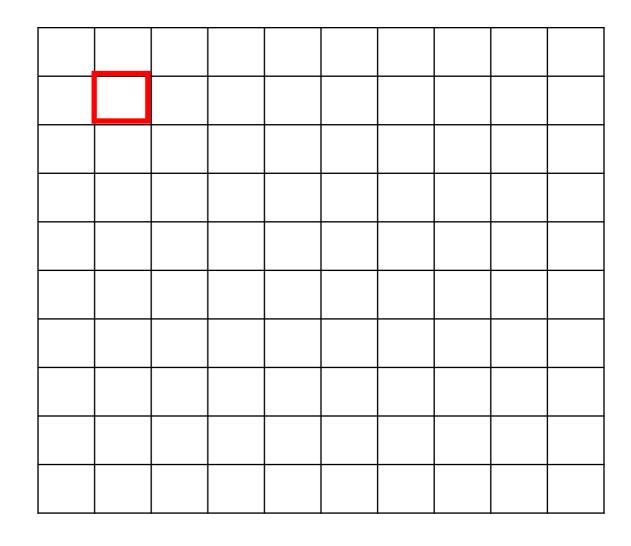
Convolution



Ex...

1	1	1	1
9	1	1	1
7	1	1	1

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	0	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0



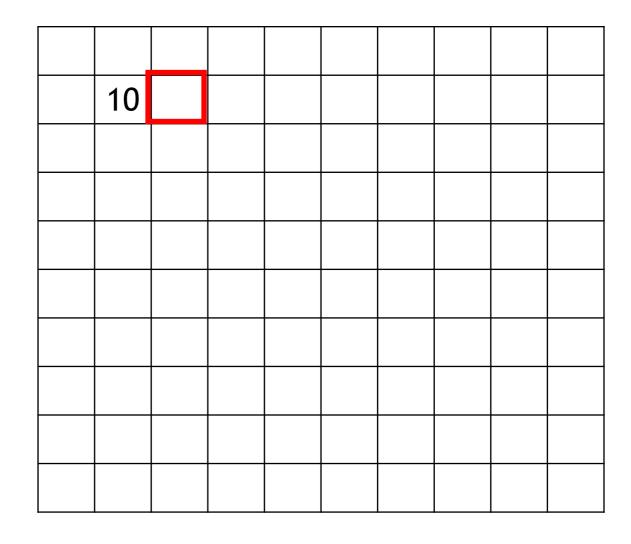
1	1	1	1
9	1	1	1
7	1	1	1

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	0	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

10				

1	1	1	1
9	1	1	1
7	1	1	1

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	0	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0



1	1	1	1
9	1	1	1
	1	1	1

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	0	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

10	20				

1	1	1	1
9	1	1	1
7	1	1	1

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	0	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

10	20				

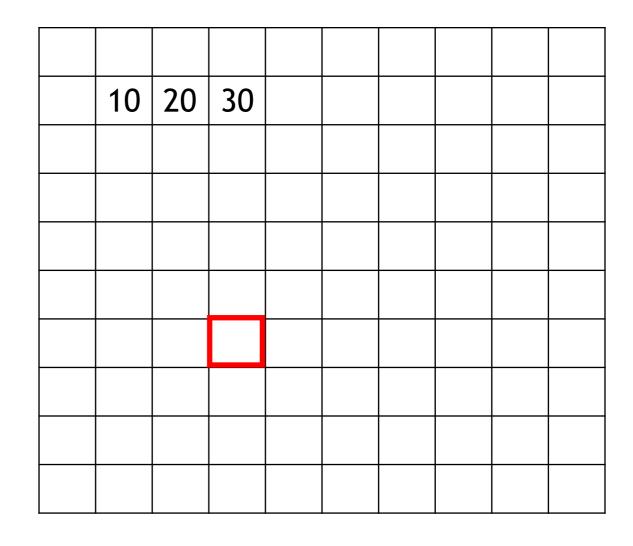
1	1	1	1
9	1	1	1
7	1	1	1

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	0	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

10	20	30			

1	1	1	1
9	1	1	1
	1	1	1

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	0	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0



1	1	1	1
9	1	1	1
7	1	1	1

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	0	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

10	20	30			
		80			

1	1	1	1
9	1	1	1
•	1	1	1

		90	90	90	90	90			
		90	90	90	90	90			
		90	90	90	90	90			
		90	90	90	90	90			
		90		90	90	90			
		90	90	90	90	90			
		0	0	0	0	0			
0	0	0	0	0	0	0	0	0	0

10	20	30	30	30	20	10	
20	40	60	60	60	40	20	
30	60	90	90	90	60	30	
30	50	80	80	90	60	30	
30	50	80	80	90	60	30	
30	50	80	80	90	60	30	
20	30	50	50	60	40	20	
10	20	30	30	30	20	10	

So how convolution can help us?

#### Sobel Filters

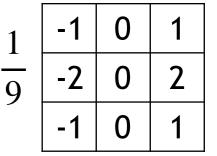
- Sobel filters create images that emphasize edges.
- Two small convolution filters are used successively:

-1	0	1
-2	0	2
-1	0	1

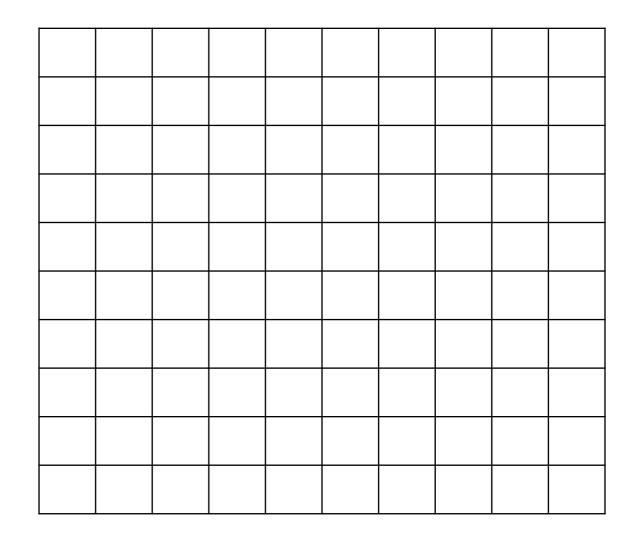
1	2	1
0	0	0
-1	-2	-1

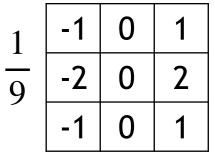
Vertical

Horizontal

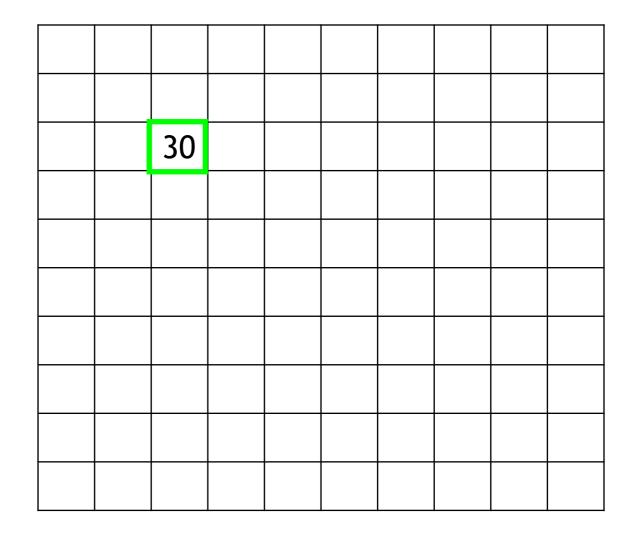


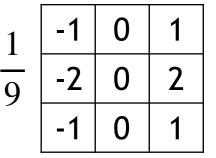
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	0	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0



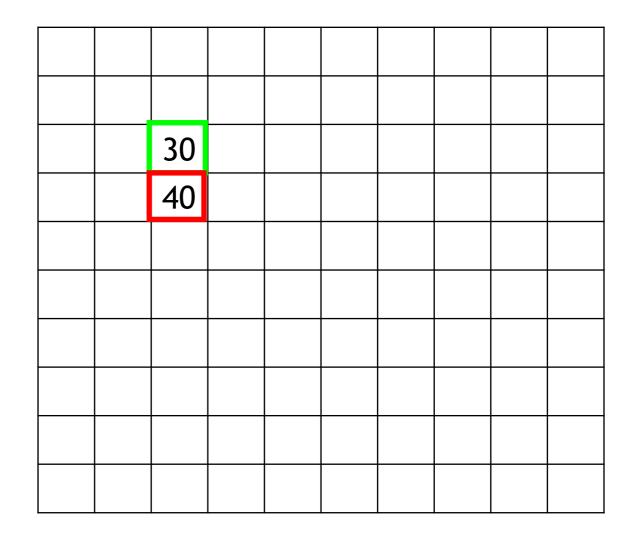


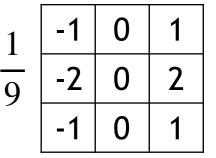
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	0	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0





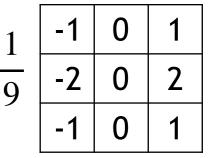
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	0	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0





0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	0	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

30				
30 40 40				
40				

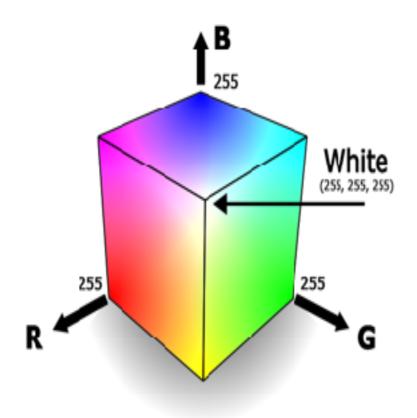


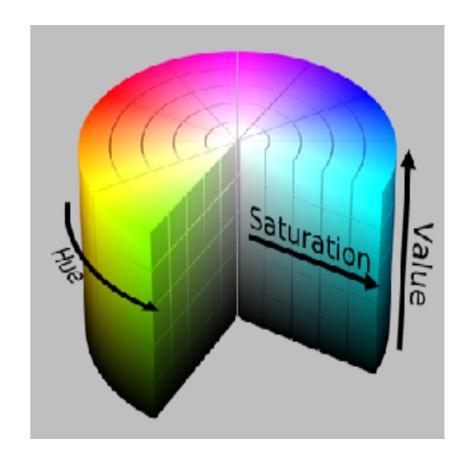
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	90	0	90	90	90	0	0	0
0	0	90	90	90	90	90	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

30				
40	0			
30 40 40				

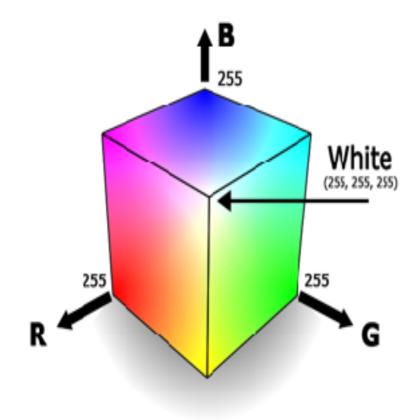
# **HSV Color Space**

## **HSV Color Space**

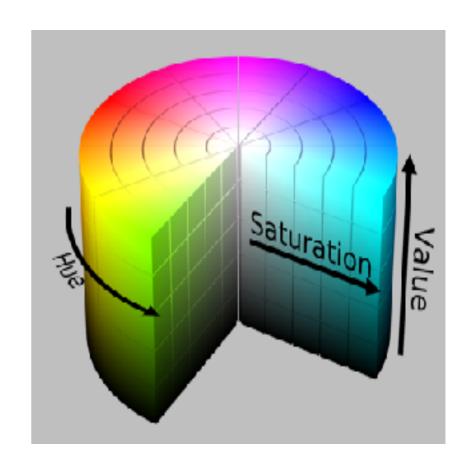




## **HSV Color Space**



(R, G, B)



(H, S, V)

**Hue Saturation Value** 

Let's go back to Jupiter Notebook!

# Google is your friends <3

