

Turtle Tkinter Matplotlib Python Imaging Library Pyglet Python Numpy Pandas Python Database

How to use the mouse to scale and rotate an image in PyGame?

Last Updated: 28 Apr, 2021

In this article, we will discuss how to transform the image i.e (scaling and rotating images) using the mouse in Pygame.

Approach

Step 1: First, import the libraries Pygame and math.

```
import pygame
import math
from pygame.locals import *
```

Step 2: Now, take the colors as input that we want to use in the game.

```
color_1 = #RGB value of color 1
color_2 = #RGB value of color 2
color_n = #RGB value of color n
```

Step 3: Then, initialize the constructor

```
pygame.init()
```

Step 4: Set the dimensions of your GUI game.

```
w, h = #Width dimension, #Height dimension
screen = pygame.display.set_mode((w, h))
```

Step 5: Set the running value for running the game, the angle by which it can be moved.

```
running = True
angle = 0
```

```
scale = 1
```

Step 6: Next, take the image as input which we want to move with the mouse

```
img_logo = pygame.image.load('#Enter the image url')
img_logo.convert()
```

Step 7: Draw a border around an image.

```
rect_logo = img_logo.get_rect()
pygame.draw.rect(img_logo, color_1, rect_logo, 1)
```

Step 8: Locate the center of the GUI game and get the position of the mouse.

```
center = w//2, h//2
mouse = pygame.mouse.get_pos()
```

Step 9: Store the image in a new variable and construct a rectangle around the image.

```
img = img_logo
rect = img.get_rect()
rect.center = center
```

Step 10: Set the things which you want your app to do when in running state.

```
while running:
    for event in pygame.event.get():
```

• **Step 10.1:** Once the app is in a running state, make it quit if the user wants to quit.

```
if event.type == QUIT:
    running = False
```

• **Step 10.2:** In case, the user doesn't want to quit, set at what angle the image should rotate.

```
if event.type == KEYDOWN:
    if event.key == K_ra:
```

```
if event.mod & KMOD_SHIFT:
    # angle at which it should move left
    angle -=
else:
    # angle at which it should move right
    angle +=
```

• **Step 10.3:** Also, set at what ratio the image size should decrease or increase.

```
elif event.key == K_sa:
    if event.mod & KMOD_SHIFT:
        scale /= #scale at which the image size should
decrease
    else:
        scale *= #scale at which the image size should
increase
```

• **Step 10.4:** Set at what coordinates, angle, and scale the image will rotate or resize.

```
elif event.type == MOUSEMOTION:
```

• **Step 10.4.1:** Store the current position of the event in the new variable.

```
mouse = event.pos
```

• **Step 10.4.2:** Locate the Cartesian coordinates with the help of the mouse variable and center of the image for rotating the image..

```
x = mouse[0] - center[0]
y = mouse[1] - center[1]
```

• Step 10.4.3: Further, calculate the distance between the two points (0,0) and (x, y) with the help of formula $\sqrt{x^2+y^2}$

$$d = math.sqrt(x ** 2 + y ** 2)$$

• **Step 10.4.4:** Now, calculate the angle in degrees at which the image should rotate using the Python method **math.atan2()** which returns the arctangent of y/x, in radians.

```
angle = math.degrees(-math.atan2(y, x))
```

• **Step 10.4.5:** Calculate which scale the image size should decrease or increase using the function <u>abs</u>, which returns the magnitude of the number.

```
scale = abs(5 * d / w)
```

• **Step 10.4.6:** Calculate the updated position of the image in the running state using the rotozoom function which is a combined scale and rotation transform.

```
img = pygame.transform.rotozoom(img_logo, angle, scale)
```

• Step 10.4.7: Construct the rectangle around the updated image

```
rect = img.get_rect()
rect.center = center
```

Step 11: Next, you need to set the screen color and the image on the screen.

```
screen.fill(color_3)
screen.blit(img, rect)
```

Step 12: Later on, draw the <u>rectangle</u>, <u>line</u>, and <u>circles</u> which will help in moving the image.

```
pygame.draw.rect(screen, color_2, rect, #thickness of rectangle)
pygame.draw.line(screen, color_3, center, mouse, #Thickness of line)
pygame.draw.circle(screen, color_1, center, #radius of circle,
#thickness of circumference)
```

```
pygame.draw.circle(screen, color_2, mouse, #radius of circle,
#thickness of circumference)
```

Step 13: Furthermore, update the changes done in the GUI game.

```
pygame.display.update()
```

Step 14: Finally, quit the GUI game.

```
pygame.quit()
```

Below is the implementation.

Python

```
# Python program to transform the
# image with the mouse
#Import the libraries pygame and math
import pygame
import math
from pygame.locals import *
# Take colors input
RED = (255, 0, 0)
BLACK = (0, 0, 0)
YELLOW = (255, 255, 0)
#Construct the GUI game
pygame.init()
#Set dimensions of game GUI
w, h = 600, 440
screen = pygame.display.set_mode((w, h))
# Set running, angle and scale values
running = True
angle = 0
scale = 1
# Take image as input
img_logo = pygame.image.load('gfg_image.jpg')
img_logo.convert()
# Draw a rectangle around the image
rect_logo = img_logo.get_rect()
```

```
pygame.draw.rect(img logo, RED, rect logo, 1)
# Set the center and mouse position
center = w//2, h//2
mouse = pygame.mouse.get pos()
#Store the image in a new variable
#Construct the rectangle around image
img = img logo
rect = img.get_rect()
rect.center = center
# Setting what happens when game is
# in running state
while running:
   for event in pygame.event.get():
       # Close if the user quits the game
       if event.type == QUIT:
            running = False
       # Set at which angle the image will
       # move left or right
       if event.type == KEYDOWN:
            if event.key == K ra:
                if event.mod & KMOD SHIFT:
                    angle -= 5
                else:
                    angle += 5
            # Set at what ratio the image will
            # decrease or increase
            elif event.key == K sa:
                if event.mod & KMOD_SHIFT:
                    scale /= 1.5
                else:
                    scale *= 1.5
        # Move the image with the specified coordinates,
        # angle and scale
        elif event.type == MOUSEMOTION:
            mouse = event.pos
            x = mouse[0] - center[0]
            y = mouse[1] - center[1]
            d = math.sqrt(x ** 2 + y ** 2)
            angle = math.degrees(-math.atan2(y, x))
            scale = abs(5 * d / w)
            img = pygame.transform.rotozoom(img logo, angle, scale)
            rect = img.get_rect()
```

rect.center = center

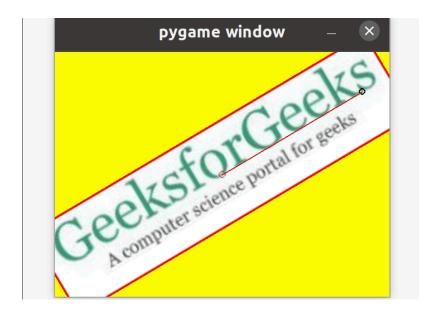
```
# Set screen color and image on screen
screen.fill(YELLOW)
screen.blit(img, rect)

# Draw the rectangle, line and circle through
# which image can be transformed
pygame.draw.rect(screen, BLACK, rect, 3)
pygame.draw.line(screen, RED, center, mouse, 2)
pygame.draw.circle(screen, RED, center, 6, 1)
pygame.draw.circle(screen, BLACK, mouse, 6, 2)

# Update the GUI game
pygame.display.update()

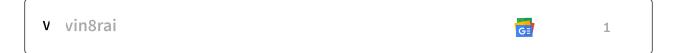
# Quit the GUI game
pygame.quit()
```

Output:



Looking to dive into the world of programming or sharpen your Python skills? Our <u>Master Python: Complete Beginner to Advanced Course</u> is your ultimate guide to becoming proficient in Python. This course covers everything you need to build a solid foundation from fundamental programming concepts to advanced techniques. With **hands-on projects**, real-world examples, and expert guidance, you'll gain the confidence to tackle complex **coding challenges**. Whether you're starting from scratch or aiming to enhance your

skills, this course is the perfect fit. Enroll now and master Python, the language of the future!



Previous Article Next Article

How to move an image with the mouse in PyGame?

How to create Buttons in a game using PyGame?

Similar Reads

How to Rotate and Scale images using PyGame?

In this article, we are going to see how to Rotate and Scale the image. Image Scaling refers to the resizing of the original image and Image Rotation refers to...

3 min read

How to move an image with the mouse in PyGame?

Pygame is a Python library that is used to create cross-platform video games. The games created by Pygame can be easily run through any of the input device...

4 min read

Adding Collisions Using pygame.Rect.colliderect in Pygame

Prerequisite: Drawing shapes in Pygame, Introduction to pygame In this article, we are going to use pygame. Rect. colliderect for adding collision in a shape usin...

3 min read

PyGame Set Mouse Cursor from Bitmap

In this article, we are going to see how to set the mouse cursor from bitmap using the PyGame module in Python. What is PyGame?It is a cross-platform set of...

5 min read

Mouse Clicks on Sprites in PyGame

The interactiveness of your game can be significantly increased by using Pygame to respond to mouse clicks on sprites. You may develop unique sprite classes th...

3 min read

Rotate a picture using ndimage.rotate Scipy

Prerequisites: Mathplotlib, Scipy Some of the most common tasks in image processing are displaying images, Basic manipulations, Image filtering, Image...

2 min read

Getting width and height of an image in Pygame

Prerequisites: Pygame To use graphics in python programs we use a module called Pygame. Pygame provides high functionality for developing games and...

3 min read

How to invert colors of an image in pygame?

In this article, we are going to learn how to invert the colors of an image using Pygame in Python programming language. Pygame is a multiplatform Python...

7 min read

Pygame - Flip the image

In this article, we are going to see how images can be flipped using Pygame. To flip the image we need to use pygame.transform.flip(Surface, xbool, ybool)...

2 min read

How to Use Multiple Screens on PyGame?

In this article, we are going to know how to use multiple screens on PyGame. Multiple screens are one of the best features of so many apps and games. This...

9 min read

Article Tags: Python Python-PyGame

Practice Tags: python



Corporate & Communications Address:-A-143, 9th Floor, Sovereign Corporate
Tower, Sector- 136, Noida, Uttar Pradesh
(201305) | Registered Address:- K 061,
Tower K, Gulshan Vivante Apartment,
Sector 137, Noida, Gautam Buddh
Nagar, Uttar Pradesh, 201305





Company

About Us

Legal

In Media

Contact Us

Advertise with us

GFG Corporate Solution

Placement Training Program

GeeksforGeeks Community

DSA

Data Structures

Algorithms

DSA for Beginners

Basic DSA Problems

DSA Roadmap

Top 100 DSA Interview Problems

DSA Roadmap by Sandeep Jain

All Cheat Sheets

Web Technologies

HTML

(55

JavaScript

TypeScript

ReactJS

NextJS

Bootstrap

Web Design

Languages

Python

Java

 \mathbb{C}^{++}

PHP

GoLang SQL

R Language

Android Tutorial

Tutorials Archive

Data Science & ML

Data Science With Python

Data Science For Beginner

Machine Learning

ML Maths

Data Visualisation

Pandas

NumPy

NLP

Deep Learning

Python Tutorial

Python Programming Examples

Python Projects

Python Tkinter

Web Scraping

OpenCV Tutorial

Python Interview Question

Django

Computer Science

DevOps

Operating Systems Git
Computer Network Linux

Database Management System AWS
Software Engineering Docker
Digital Logic Design Kubernetes
Engineering Maths Azure
Software Development GCP

System Design

Software Testing

High Level Design
Low Level Design
UML Diagrams
Interview Guide
Design Patterns

System Design Bootcamp
Interview Questions

OOAD

Inteview Preparation

DevOps Roadmap

Competitive Programming
Top DS or Algo for CP
Company-Wise Recruitment Process
Company-Wise Preparation
Aptitude Preparation
Puzzles

GeeksforGeeks Videos

CS Subjects

School Subjects

Mathematics DSA Physics Python Chemistry Java Biology C++ Social Science Web Development English Grammar Data Science

Commerce World GK

@GeeksforGeeks, Sanchhaya Education Private Limited, All rights reserved