

# NerdParadise

Digitally re-remastered anniversary collection

[Forum](#)[About](#)[FAQ](#)[Store](#)[NP](#) > [Technology](#) > [Python](#) > [PyGame](#) >

## How to Blit Images With Transparency at Lower Opacity in PyGame

Tags: [Python](#), [PyGame](#), [PyWeek](#)Post by: [Blake](#)

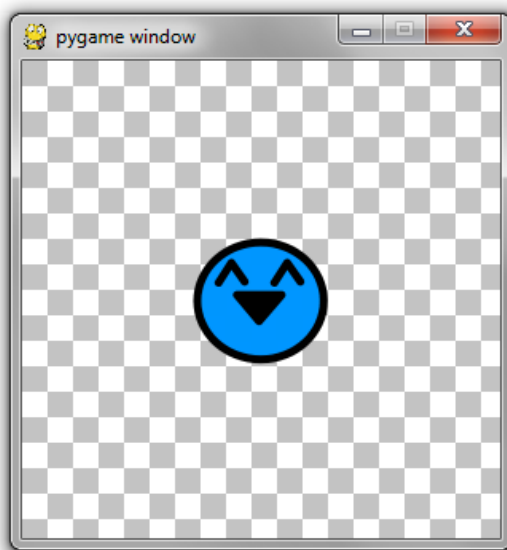
Posted on: 11 Cresco 13:3 - 8.35.7

PyWeek started today. So I'll take a little detour from the regularly scheduled PHP tutorial and give a little tidbit that some of you may find useful if you're using PyGame.

If you've used PyGame for any amount of time, you may have noticed that it's impossible to (directly) blit an image with per-pixel transparency at a non 100% opacity. So if you have a sprite image that has transparent pixels around the edges (as most sprite images do) and you want it to fade out, you're sort of out of luck.

Luckily, there's a creative hack you can do.

Let's look at a simple "game" where there is a sprite with per-pixel transparency in front of some background...



The code for this game is as follows:

```
import pygame
import time

pygame.init()
screen = pygame.display.set_mode((300, 300))
done = False
```

[Browse](#)[Latest Posts](#)[Most Popular](#)[Random](#)[Root Nodes](#)[Technology](#)[Puzzles](#)[Math](#)[Silly](#)[North Avenue](#)[Science](#)[History](#)[Literature](#)[Origami](#)[Random Tags](#)[Rubik's Cube](#)[Silverlight](#)[Math Tricks](#)[Game Programming](#)[Python](#)[Silly](#)

```
happy = pygame.image.load('happy.png') # our happy blue protagonist
checkers = pygame.image.load('background.png') # 32x32 repeating checkered
image

while not done:
    start = time.time()
    # pump those events!
    for e in pygame.event.get():
        if e.type == pygame.QUIT:
            done = True

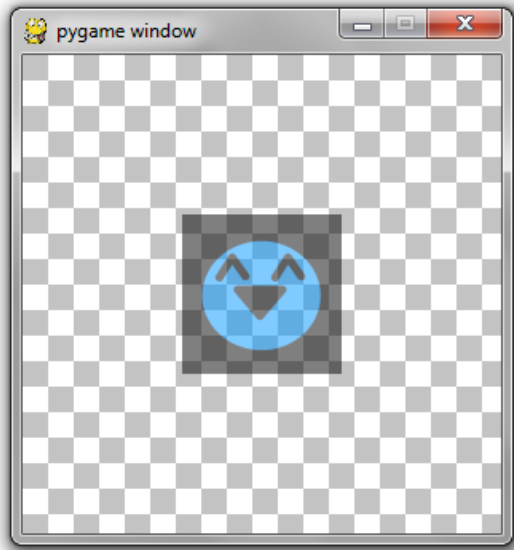
    # checker the background
    x = 0
    while x < 300:
        y = 0
        while y < 300:
            screen.blit(checkers, (x, y))
            y += 32
        x += 32

    # here comes the protagonist
    screen.blit(happy, (100, 100))

    pygame.display.flip()

    # yeah, I know there's a pygame clock method
    # I just like the standard threading sleep
    end = time.time()
    diff = end - start
    framerate = 30
    delay = 1.0 / framerate - diff
    if delay > 0:
        time.sleep(delay)
```

The problem here is that the "happy" Surface contains alpha per pixels. If you wanted to fade the sprite out, you'd probably try to use a `happy.set_alpha(new_alpha)`, however, `.set_alpha` does not work on per-pixel transparency. If you convert happy to not have per-pixel alpha, then you get an ugly box around it...

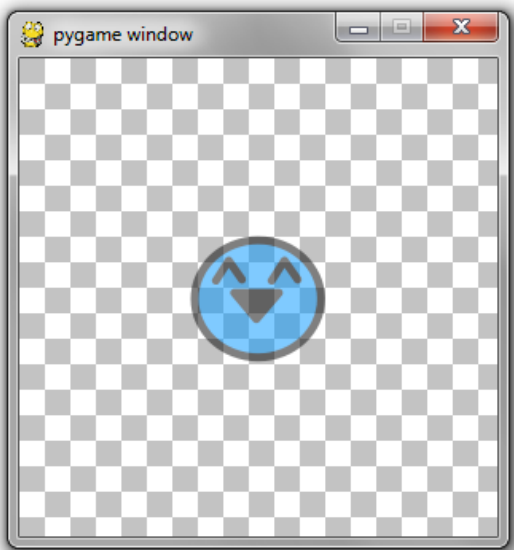


So that's not the solution either.

So I present to you a little hack...

```
def blit_alpha(target, source, location, opacity):  
    x = location[0]  
    y = location[1]  
    temp = pygame.Surface((source.get_width(), source.get_height())).convert  
    temp.blit(target, (-x, -y))  
    temp.blit(source, (0, 0))  
    temp.set_alpha(opacity)  
    target.blit(temp, location)
```

If you replace the `screen.blit(happy, (100, 100))` with a call to `blit_alpha(screen, happy, (100, 100), 128)`, you get the following:



How it works:

- Create a temporary image OPAQUE the size of the image you are trying to blit.
- Blit the OPAQUE BACKGROUND onto this temporary image.
- Blit the per-pixel transparency image onto the temporary image.
- The temporary image is completely opaque and has the transparent image on it above the background.
- Because the temporary image is opaque, you can set the image alpha of it.
- Blit this image onto the background at the desired alpha.

Perfect!

Well, almost...

Because I'm creating a temporary surface per frame, this is not a very performant solution. Ideally, you should keep a cache of temporary surfaces. A dictionary with string keys in the format of "[width] x [height]" will work. If the cache contains the size image you need, re-use it. You don't have to worry about what was on the image previously since you'll be blitting over it anyway. This assumes that the images you will be blitting with alpha are usually the same ones over and over.

Happy PyGaming and/or PyWeeking!



#### User Comments: 1

Post by [coppolaemilio](#) on 12 Ineo 4:2 - 5.37.94

Thank you very much for this tutorial, do you want credit for the code?

Best Regards

You must be logged in to add a comment

Current Date: 12 Ineo  
7:3

Current Time:  
14.60.87

Your IP: 70.90.14.5

Browser: Firefox

Browser Version:  
12

Join us in IRC...  
Server: [irc.esper.net](#)  
Channel:  
[#nerdparadise](#)

© 2012 Nerd Paradise