



# Raspberry Pi Photo Booth

 June 15, 2018  [Featured](#), [Raspberry Pi](#)

A DIY photo booth for events that takes photos and uploads an animated gif to a Tumblr account.

## Now Version 3.0

I recently updated this tutorial and code for a new version. It's simpler and works more efficiently. [Also, I added an optional second raspberry pi connected to a projector showing the event's images.](#)

## Quickstart

If you are just looking for the nitty gritty, consider looking at the [code on GitHub](#), the [how to photos on Flickr](#), and the [results on Tumblr](#). You will also need [picamera](#), [pyTumblr](#), and [GraphicsMagick](#). Be sure to have a strong connection to the internet (wired is best) and good lighting. Set yourself up for success with the photos and posting.

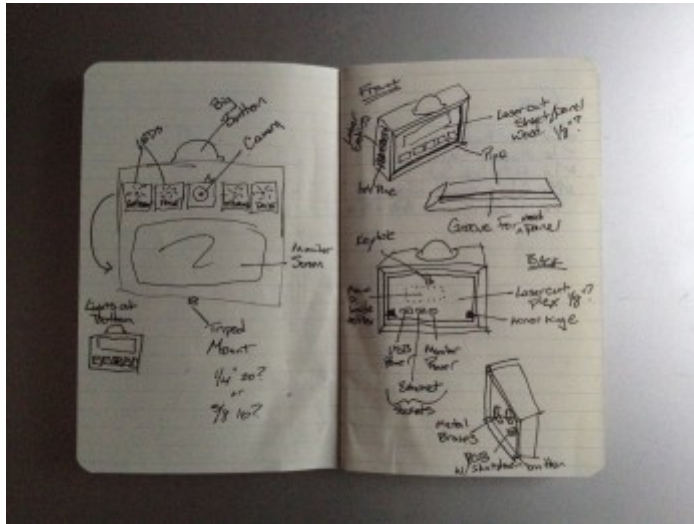
## Intro

This post begins with a story. My wife started grad school in the Fall of 2012 at the [Cooperstown Graduate Program](#). We moved to Cooperstown, NY in the summer of 2012. Given the two-year program, it gave me a two-year duration for a photo project. I decided to use the iPhone app [PocketBooth](#) to take photo strips of people that we would meet in those two years. Of course this was before the word "selfie" was used all the time. It turned into a fun project. I printed the strips and hung them in our house as the number grew. Then, on the night of graduation, I brought the exhibit to their reception. It went over very well. I then [posted the photos online on Flickr](#).

Overall I was happy with the project. I was constantly surprised at how easy it was to get people to smile and let loose in front of a camera when it's a photo booth. However, the overall project was very time intensive. I had to take my phone out of my pocket, open the app, take the photo strips, save the strips to my computer, print out the strips, cut the strips, and then hang them on the exhibit. Part of the beauty of the [original photo booths](#) is they were automated.

So then I decided to make an automated photo booth for my wife's graduation reception—a bit of a guerrilla exhibit. It sounded like fun. I had been playing around with the [Raspberry Pi](#) for a bit, so this seemed like a great use for the [camera module](#).

## The Plan



Original sketch of the idea

So the goal was to make an easy to use photo booth for guests at an event. Then turn the photos into an animated gif and automatically post it online for easy sharing.

I began with a notebook sketch of the idea, algorithm, and construction. See [here](#). Then I researched [other photo booths](#) out there for inspiration. Then I started wiring, coding, and testing. Once I had the parts working, then I built the enclosure.

Here is the basic algorithm:



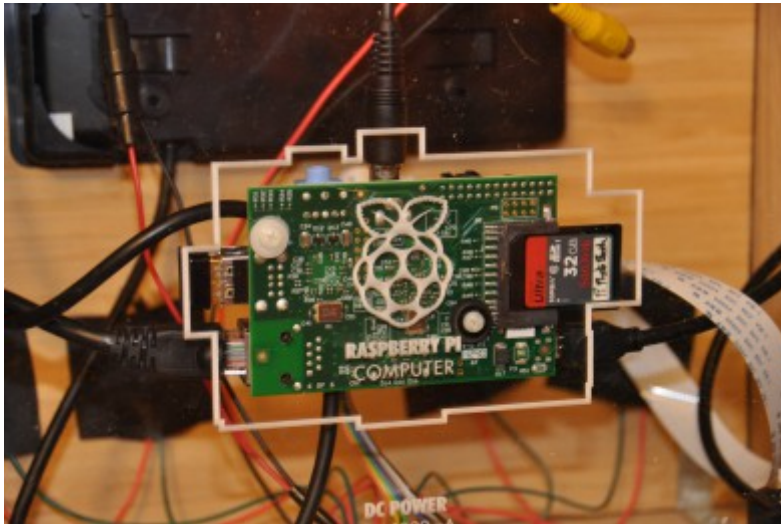
### Algorithm idea sketch

- Setup
  - Set the device on top of a camera tripod and point it at a good location for guests to stand
  - Make sure there will be good lighting on the guests
  - Power up the device
  - Light up all the lights on the front for a bit to show that it's running, then turn them off

- Await a button press
- Big button press
  - Once the button is pressed, blink the first light for a few seconds, telling the guests to get ready
  - Show the real time video preview on the screen
  - Take four pictures
  - Light up the “Pose” light (Note, in later versions, I removed the LEDs for the steps. It was a better user experience to just put images on the screen for the steps.)
  - Take a photo and save as a .jpg with a time stamp in the filename
  - Repeat four times
- Process the images
  - Turn off the preview monitor
  - Blink the “Uploading” light
  - Combine the four jpgs into a new animated gif (consider adding a footer to all the pics)
  - Upload to Tumblr (since Tumblr natively handles animated gifs)
  - Stop blinking the light
- Done
  - Light the “Done” light for a few seconds
  - Replay the images on the screen a few times for instant gratification of the guests
  - Await another button press
- Shutdown
  - Have another button in the system, and if it is pressed shutdown the Raspberry Pi safely. Light all the LEDs first as a clue to when it’s safe to turn off the system. (Note, I removed the second button. It was working inconsistently. And adding keyboard interrupts was a lot easier to use when coding. It may be a bad habit to just turn off the raspberry pi instead of a proper shut down. But I have yet to have a corrupted card. Cross my fingers I don’t have problems. If you’re worried, just plug in a keyboard, hit escape to exit pygame, hit ctrl-c to exit python, and then type sudo halt to shutdown.)

## The Parts





View from the rear showing the guts, including the Raspberry Pi

Here is a basic parts list without the enclosure:

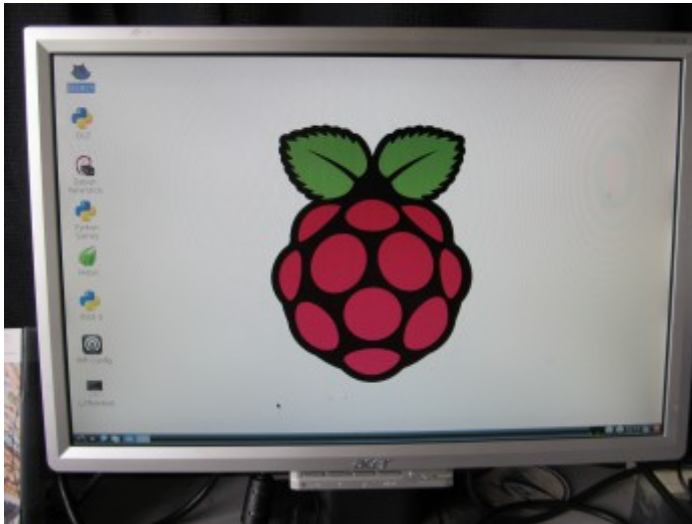
- **Raspberry Pi** computer ~\$40
  - The newer pi's work just fine too.
- **Raspberry Pi camera module** ~\$30
  - I'm using a version 1 camera, but newer versions work as well.
- SD Card
  - I used a **32 GB card** with a decent read/write speed
  - But an old camera card at least 8GB should be fine
- **5V Power adapter with USB A to Micro B cable** ~\$10
- 7 inch monitor
  - I used a **low resolution car backup monitor** ~\$30
  - But you may want to consider a higher quality monitor that is easier to use. **Adafruit sells a nice HD one. Or a SD one.**
- Video cable
  - Either **HDMI** (\$6) or **RCA** (\$7) depending on what type of monitor your use. The shorter the better. I ended up trimming down an old RCA cable for my purposes.
- Video power adapter
  - You'll need one unless it's included in your monitor purchase. I used a **variable adapter** set to 12v that I had sitting around.
- Keyboard
  - A standard USB keyboard you have laying around is just fine. You'll only need it during configuration, not during usage.

- I use at [wireless keyboard/mouse combo](#), which is nice to leave plugged in even during operation at an event.
- Mouse
  - A standard USB mouse is fine, as you'll just need it during configuration.
- An arcade button. I used one with a built in 5V LED ~\$2
- [GPIO cables](#) ~\$10
  - Used to connect the GPIO pins to the button
- A tripod mount
  - Originally I made one to fit my camera tripod, but it was a little too shaky.
  - On the latest version, I used a [PA system tripod mount](#). It's much more stable.

Here is an optional parts list, some things that are really nice to add:

- [WiFi adapter](#) ~\$12
- [TTL cable](#) ~\$10
  - If you want to trouble-shoot at an event, these make it really easy to use your laptop to change things on the pi. [See these instructions](#).
- If you want to modify the circuit design at all, a [Pi Cobbler](#), [solderless breadboard](#), and [jumper wires](#)
- [USB power panel mount](#) ~\$4
  - It's nice in the finished product to unplug the power and not have to reach into the enclosure. This is used to power the Raspberry Pi. If you use it, then you'll also need a standard [USB A to B cable](#). You probably have one sitting around.
- [Ethernet panel mount](#) ~\$5
  - If you're using a wired connection, which is nice for speed and reliability, then this makes it much easier to connect in the final enclosure. You'll also need a longer Cat 5 or Cat 6 cable.
- [DC power jack](#) ~\$4
- [USB port panel mount](#) ~\$4
  - I didn't add this, but I wish I did. It would make it easier to troubleshoot without having to open the case.
- [Extension cable for the camera module](#) ~\$5

## Setup the Raspberry Pi



Raspberry Pi Graphical Interface — Photo by [Gary Higgins](#)

If you are new to the Raspberry Pi, there are lots of [good tutorials to get](#) you started. But here are some quick links on what's needed for this project.

Install [noobs](#) on the SD card.

[Setup the WiFi.](#)

If you are using the [Bluetooth keyboard and mouse combo](#), [here are some instructions for setup.](#)

Somehow you'll want to move files to and from the Raspberry Pi. [If you have a Mac, this is the easiest way.](#) [This looks good on a PC](#), but I've never tried it.

I'd recommend doing most of the configuration using a large external monitor. When you're ready to move to the small screen, note that I ran into errors getting the composite video working. I found that I needed to [edit a config file](#).

```
sudo nano /boot/config.txt
```

Comment out the line

```
# hdmi_force_hotplug=1
```

I ended up figuring out some [other settings](#) to get the video looking decent on the low resolution monitor.

```
start_x=1  
gpu_mem=128
```

```
framebuffer_width=800  
framebuffer_height=480  
sdtv_aspect=3  
disable_camera_led=1
```

If you haven't already, [setup the Raspberry Pi to automatically boot into the GUI desktop](#).

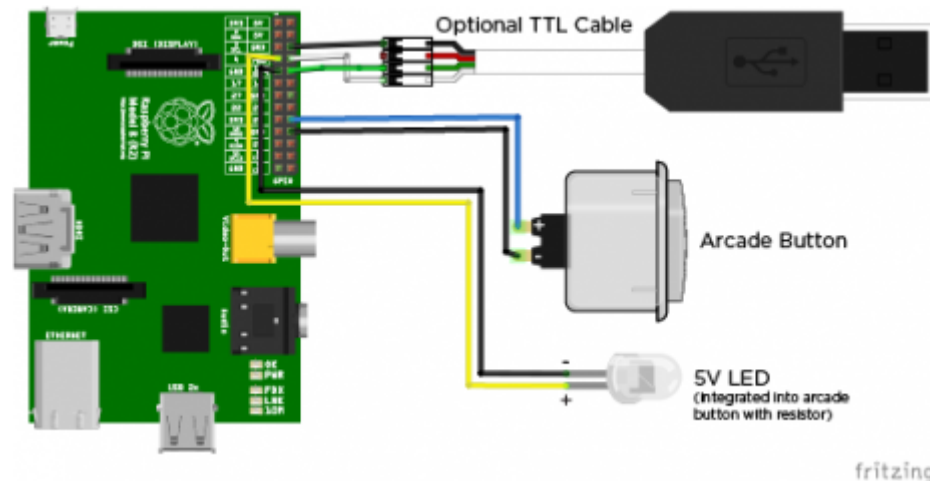
[Disable the screensaver](#) by editing the GUI session autostart.

```
sudo nano ~/.config/lxsession/LXDE-pi/autostart
```

Make sure the following is in the autostart file:

```
@xset s 0 0  
@xset s noblank  
@xset s noexpose  
@xset dpms 0 0 0
```

## GPIO, LEDs, and Buttons



Basic GPIO circuit

One of the cool things about the Raspberry Pi is you can wire your own inputs and outputs. This is all done through the [GPIO](#). I first worked out the circuit using a solderless breadboard. You can see the [wiring diagram](#) below. Then I got my wire cutters, soldering iron, and heat shrink tubing out. I used the gpio cables to connect to the button.



# Taking Photos



Raspberry Pi Camera Module and laser etched front face plate from version 1 of the photo booth

First, with the system turned off, plug-in the camera. Then boot it back up. You'll need to follow the [initial setup instructions](#).

I decided to use [PiCamera](#) to control the camera with Python. Install [PiCamera](#).

## Creating the Animated Gif

I first tried [ImageMagick](#), but had [some issues](#) with the jpg delegate. I then used [GraphicsMagick](#) instead. It works great. [Read the installation instruction here for Unix](#). But the following should work.

```
sudo apt-get install graphicsmagick
```

Once GraphicsMagick is installed, you can test it with the following code. It will combine all the jpgs in the current directory.

```
gm convert -delay 100 *.jpg joined.gif
```

## Uploading to Tumblr

I used [pytumblr](#) and the [tumblr API](#) to get the photos from the pi to the internet.

To install pyumblr, follow these steps:

1. If it is not installed on the pi, [install PIP somewhere on your pi](#)
2. In the terminal, navigate to the pip folder you downloaded and moved to the pi. Then run the following code:

```
sudo python get-pip.py
```

3. Download [pyumblr](#) and copy the files somewhere on your pi. For instance, in the command line, you could navigate to your chosen file and then type:

```
wget "https://github.com/tumblr/pyumblr/archive/master.zip"
```

4. In the terminal, navigate to pyumblr-master folder (you [un-zipped](#) the downloaded file, right?) and run this command to install:

```
sudo python setup.py install
```

## The Final Code

```
1  #!/usr/bin/env python
2  # created by chris@drumminhands.com
3
4  import os
5  import time
6  from time import sleep
7  import RPi.GPIO as GPIO
8  import picamera
9  import atexit
10 import sys
11 from email.MIMEText import MIMEText
12 from email.MIMEImage import MIMEImage
13 from email.MIMEBase import MIMEBase
14 from email import Encoders
15 import smtplib
16 import socket
17 import pygame
18 from signal import alarm, signal, SIGALRM, SIGKILL
19
20
21 #####
22 ### Variables Config ###
23 #####
24 led1_pin = 15 # LED 1
25 led2_pin = 19 # LED 2
26 led3_pin = 21 # LED 3
27 led4_pin = 23 # LED 4
28 button1_pin = 22 # pin for the big red button
29 button2_pin = 18 # pin for button to shutdown the pi
30 button3_pin = 16 # pin for button to end the program, but not shutdown the pi
31
```

[Download the photo booth code here](#)

You can download [the code at GitHub](#). If you have not used GitHub before, [check out this link](#). There are a few values for variables that you will need to replace with your special information. Save the code wherever you want. I put it in the folder named /home/pi/photobooth/. If you are not comfortable using the command line, [learn more here](#).

After getting the photobooth code on your Raspberry Pi, set it to [automatically run the python code at startup](#).

```
sudo nano /etc/xdg/autostart/myscript.desktop
```

And save something like the following:

```
[Desktop Entry]
Type=Application
Name=drumminhands_photobooth.py
Comment=Start my script
NoDisplay=false
Exec=sudo python /home/pi/photobooth/drumminhands_photobooth.py
NotShowIn=GNOME;KDE;XFCE;
Name[en_US]= drumminhands_photobooth.py
```

I found I need sudo at the beginning, otherwise the GPIO buttons and lights don't work.

## Enclosure



Photo booth enclosure



Case for Version 3 Photo Booth

In the first versions, I built my own case out of wood. They were a little rough as I'm still working on my carpentry skills. I did laser cut the front and back plates.

In the latest version, I decided to re-use some existing items to make a cooler looking case. I found a small brief case at an antique shop. It was originally used for holding silverware. Then I also re-purposed an old camera I had. I took off the front plate and the lens. I used an existing hole in the front plate for the new LED arcade button.

If you design your own cover, you'll need the [dimensions of the camera](#), [Raspberry Pi mount](#), [the Ethernet mount](#), and [the USB mount](#). Check the [github site](#) for my front and back templates.

## Lessons Learned

- It works best with internet access. When the users can instantly use their phone to share photos from Tumblr, it's an overall better user experience.
- Use wired access if possible. Wireless is sometimes flaky. And using a mobile hotspot is even more sketchy. Without a good internet connection, the experience is a little bit of a drag.
- There is an option to not upload during the event. The replay of pics on the photo booth does give the immediate gratification to the users who just had their photos taken. But it does not show the rest of the guests, nor friends not at the event. I did recently [build another tutorial to have a second raspberry pi showing a slideshow of photos](#). That's a good second if you don't have internet access. And I also built a [batch upload script](#) if you want to put photos on Tumblr after the event.
- Use LOTS of light. I mean LOTS. Set yourself up for success. Also, look into upping the ISO on the camera if you know light will be poor. At the last event I used this at, some of the photos were taken just before sunset. The ambient light combined with overhead shop lights made

much sharper images. At night, with only the shop lights, there were still lots of blurry photos.

- Consider the height of your audience. For many adult audiences, this works quite well on a strong tripod. But kids and other shorter individuals have a tough time getting their faces into the frame without a step, bench, or ramp. At the last event, I tried using a loveseat and lowering the camera. It worked fairly well, but when people were seated, they could not reach the button. If they stand, then they can step backward after hitting the button before the first shot.
- I might look into using a version of Linux that boots up faster. If the pi loses power during the event, you need to explain what is happening during the reboot.
- I might build a better power circuit so that there is only one power adapter.

## Enjoy

See my final pics here: [drumminhands.tumblr.com](https://www.drumminhands.com). If you build one, post your link in the comments below. If you have any questions about this, also post in the comments below. Have fun with it.

And if you make one similar, please send me photos or post them in the comments below. I'd love to see other iterations in use.

## Iterations

See what other folks have done:

- [St. Louis Science Center](#) had their [ScienceTechGirls](#) make [one](#).
- [@jspotproduction](#) made one with [a great CNC enclosure](#).
- [@CongoJoe](#) made [one for a wedding](#).
- Here's [another](#).

## Press

Some folks have been talking about this project. Here are a few. If you have a post about the project, please let me know and I'll add it here.

- [CNET](#)
- [LifeHacker](#)
- [Adafruit](#)
- [Makersify](#)
- [IT Clips](#)





## Search

Search site...

GO

## Navigation

Portfolio

Contact



## 141 Comments

Join the discussion and tell us your opinion.

Eoghan [reply](#)

April 17, 2016 at 4:26 pm

Thanks very much for taking the time to post your detailed instructions Chris!

It inspired me to create a multi-function photo booth that can:

- Take portrait photos, and crop/resize them to set sizes.
- Take photos with an accompanying overlay image.
- Take an animated gif, as yours does (but uploaded to a separate server instead of Tumblr)
- Take continuous photos (comparing with a base image to decide which to upload)

I've detailed the process at:

<http://eoghan.me.uk/notes/2016/03/28/photo-booth/>

It's not as nice and portable as yours, but will hopefully be useful at work.

Thanks again.

**Mark** [reply](#)

April 18, 2016 at 5:55 am

Thanks for this tutorial.

I've build one for my wedding next month (27th of May)!

<http://partypics.tumblr.com/>

I did several changes on the code on github

[https://github.com/galadril/drumminhands\\_photobooth](https://github.com/galadril/drumminhands_photobooth)

Here is how the result looks:

<http://partypics.tumblr.com/fotobox>

**Mark** [reply](#)

— May 28, 2016 at 6:58 am – In reply to: [Mark](#)

Damn..

Our photobooth was a BIG hit!!

<http://partypics.tumblr.com>

Thanks for this tutorial !!

**Matt** [reply](#)

— June 2, 2016 at 6:47 am – In reply to: [Mark](#)

Hi mark, I really like your changes that you made particularly the countdown timer that I'm trying to add to my own design. I've tried your code but it seems to freeze half way through? Any idea why?

Cheers

Matt

**Jeff** [reply](#)

— June 16, 2016 at 11:53 am – In reply to: [Mark](#)

Do you have a video of this running? It looks like you fixed a couple of the issues I was encountering. I would be interested in seeing it in action!

**Neill** [reply](#)

July 27, 2016 at 5:43 am

Thanks for the sharing,

in a reciprocal fashion, try using a 'top hat' PA speaker socket and PA stand instead of a tripod.

I used it in my 3ft photobooth camera... a lot more stable..<https://www.flickr.com/photos/bosscock/14481308918/in/dateposted-public/>)

Thanks again..

Neill

**Julien** [reply](#)

July 29, 2016 at 4:57 am

Hello,

It's a really nice project and a great tutorial. I'll try to build the same one but I hope I can do one or two modifications.

Can I imagine to store all the picture on one USB key or Hard drive in JPEG files + Upload on tumblr as a GIF ?

Did any one sync a LED light strip with the camera? If yes, how did you do it?

One more time, thank you very much for this tutorial.

drumminhands [reply](#)

— July 29, 2016 at 8:57 am – In reply to: [Julien](#)

You can store the images locally on the RPi, or potentially to a USB jump drive or hard drive. You may want to tweak the code to take a larger photo than it currently does (as right now it's just taking a web quality image to upload.). Then when you make the animated gif, you can resize prior to merging. Should be fairly straight forward.

As for lights, I have not tried to get a pulse of light to match the camera taking a photo. I see it being fraught with timing issues. Personally I just flood the area with lots of light. I have seen others add more light in their build, but most simply have the lights turn on at the beginning, stay on during the photo taking part, and then turn off. That way the camera will always have the right light no matter when it takes the photo.

Cheers.

matt

— July 29, 2016 at 5:41 pm – In reply to: [drumminhands](#)

It is possible to enable flash via picamera by editing the firmware

<http://picamera.readthedocs.io/en/release-1.10/recipes2.html#using-a-flash-with-the-camera>

John

— August 20, 2016 at 6:59 am – In reply to: [drumminhands](#)

Enabling the flash to work with the camera works really well. Trying to sync a pulse of light with the camera isn't hard, but produces pretty rubbish results in terms of colouring and whatnot, since the camera doesn't adjust properly.

Using the built in flash command allows the camera to trigger the flash itself, and consequently you get two flashes – once to adjust the settings, and once to take the image. The link provided above is what I used to get going. I'm using a pnp transistor to drive my flash, so these are the pin settings I used.

```
pin@p21 { polarity = "active_low"; function = "output"; termination = "no_pulling"; };
```

```
pin_define@FLASH_0_ENABLE {  
  type = "internal";
```

```
number = ;  
};
```

Matt [reply](#)

— August 4, 2016 at 1:44 pm – In reply to: [Julien](#)

you can edit the firmware for Picamera to enable flash, it depeding what model pi you have. Its fairly easy

<http://picamera.readthedocs.io/en/release-1.10/recipes2.html#using-a-flash-with-the-camera>

John [reply](#)

August 3, 2016 at 10:26 am

An internet connection wasn't available and Now I have 116 gifs to upload. How can I modify the code to upload them all at once but in separate posts? I'm thinking a just need a script that loops through this part, but each time through advances the file name. Any advice?

thanks

```
file_to_upload = config.file_path + now + ".gif"
```

```
client.create_photo(config.tumblr_blog, state="published", tags=["drumminhandsPhotoBooth"], data=file_to_upload)
```

```
break
```

drumminhands [reply](#)

— August 27, 2016 at 4:56 pm – In reply to: [John](#)

I haven't tried to do this yet, but I'll put it on my list to attempt.

drumminhands [reply](#)

— September 11, 2016 at 7:31 pm – In reply to: [John](#)



Check github. I just made a file named batch\_upload.py to accomplish just this.

Shanik Tanna [reply](#)

August 21, 2016 at 10:45 pm

Is it possible to create a video booth using a similar technique? would love to have for a charity event next week.

drumminhands [reply](#)

— August 27, 2016 at 4:55 pm – In reply to: [Shanik Tanna](#)

It is possible. It would require code revisions and is most likely not possible to accomplish within a week.

Matt [reply](#)

August 29, 2016 at 5:47 am

Does anyone know how to add a logo or watermark to the gif?

drumminhands [reply](#)

— August 30, 2016 at 1:37 pm – In reply to: [Matt](#)

I would use graphicsmagick. It has a lot of options for combining images, including transparencies.

matt

— September 2, 2016 at 11:33 am – In reply to: [drumminhands](#)

Mr Drumminhand

ok i've figured out how to add a logo to an image on gm with this –

```
gm convert +shade 30x60 *.jpg wm.jpg
```

```
gm composite -compose bumpmap -geometry +30+240 logo.png wm.jpg watermarked.jpg
```

but getting it to work on your photobooth script is beyond me. Do i need to add it to the ,jpg's or to the ,gif? I've made a 'script launcher' that starts on boot that will watermark a directory but it'll only do it once.

Now I've hit a wall.....

## drumminhands

September 11, 2016 at 7:28 pm – In reply to: [matt](#)

Anyone want to help out here? Sorry, I haven't had time to look at the code yet.

## Faraday [reply](#)

September 2, 2016 at 6:31 am

Thanks! the one on Github from March 2016 Nuts and Volts Magazine doesnt work well, and theres typos in his code (I always run code through spellcheck, as many coders are math gods and language dolts) so the camera never comes on. To boot, he grounds 3v through his camera button switch – no load...lol. He used a model A, so maybe nobody caught his mistakes. Thanks for your work & details. I'd like to build yours and change it a little, have it take pics from a motion sensor before people are ready as a prank, might be funny seeing people goof around before they are "officially" goofing around.

JOHN: I dont think you'll be successful doing that even with a good method for it, as Twitter doesnt like being pounded like that. I couldnt get a tweetbot to post very fast using a smiliar stuff. I think its to prevent spam/flooding/breaking something/whatever

## juha [reply](#)

September 16, 2016 at 4:32 pm

This is awesome project, going to build one! Does this save original images and not just gifs?

First thought was to use printer to print separate images out, but secondary RPi with projector sounds pretty good and might skip the printer part 😊

My set up and the ultimate goal would be: 2x RPi + 4G mifi to upload single images (+gifs) to wherever so guests could take them from there and gifs would go to tumblr and also to projector. Is it possible easily, even if the guy behind this build will be an idiot? 😊

drumminhands [reply](#)

— September 19, 2016 at 9:19 am – In reply to: [juha](#)

A printer is a regular request for this project. I have not implemented one, so I do not know how difficult it would be. Just make sure there are drivers for the printer on linux/raspian. To your first question, yes, the code does save high res images. So, you could modify to print those instead of the animated gif.

JA [reply](#)

September 19, 2016 at 3:49 am

This is a great tutorial. It's almost building the case to go. I do have at least one question, what happens when tumblr daily upload limit has reached? Is it 150 GIF's what can be uploaded daily? Not sure will that be enough for the upcoming event.. Plan to upload original images to google drive or onedrive or wherever somehow and animated gif's goes to tumblr with your code.

drumminhands [reply](#)

— September 19, 2016 at 9:17 am – In reply to: [JA](#)

I have yet to have an issue with tumblr's limits. I'm not sure the exact number of posts we can submit. I can't find that on their documentation right now. It may be file size limits? Can anyone else answer? But I can say that at each event I use the photo booth, all posts are uploaded just fine.

JA

— September 19, 2016 at 10:47 am – In reply to: [drumminhands](#)

This is what i've seen

"Tumblr limits each account to 250 posts per day, across all of your blogs. Within those 250 posts and reblogs, you can create 150 original photo posts per day"

Daily reset for the post count happens midnight New York time. Which is not good as i live +7 hours from it, and i think party wont last for that long.. 😊

There seem to be some website where you could check how far you are from the limits, it's postlimit.com

JA [reply](#)

September 19, 2016 at 10:56 am

Quoted from the postlimit.com FAQ.

"Can I get around post limit?"

Basically no. Despite persistent rumors to the contrary, changing your time zone in your settings does not work -your time zone has nothing to do with it.

You can use the previously described submission trick for an "emergency post".

It is also possible to create a single blog to which "you" can post 500, but only by using two accounts. You create a sideblog on one account and then create an invitation for someone else to join. Then go ahead and "be that other person" i.e. go ahead and join your own blog from the second account. Now it's a group blog with "two owners". In that case you can post 250 from one account then go log in to the other account and post 250 more to the same blog. You could even extend this to three or more accounts if you really wanted to."

So i just need to create another account then if I notice that 150 limit wont be enough then quickly change the user during the event. Not sure is this workaround possible via API...

drumminhands [reply](#)

— September 19, 2016 at 11:05 am – In reply to: JA

I suppose you could keep track of how many pictures you've posted in an event, then switch to a secondary API login and keep on posting. But this seems like a miss use of the Tumblr API to me.

## Leave a reply

Logged in as [drumminhands](#). [Log out?](#)

Comment

//

Comment





## What We Do

We help exhibit experts create interactive media-based experiences so that visitors become more engaged and exhibits drive home interpretive messaging. We make visitors weep, jump for joy, point, share, and ask others to join them in their discovery. We strive to blow visitors' minds and have them bring their friends and family back for more visits.



## Who We Are

Drumminhands Design is a group of designers, coders, interpretive thinkers, visitor experience crafters, and technological magicians. Led by founder and principal Chris Evans, we offer a commitment to quality custom media-based software and hardware solutions with easy to maintain parts and pieces.



## How We Work

Unlike many focused on technology first, we build through careful messaging followed by correct media choices, driving home interpretive stories. Technology should disappear leaving a sense of awe and magic. We use principles of Universal Design to create inclusive and approachable experiences. We contribute to the overall collaborative and creative exhibit team of experts.

© Copyright 2021. All Rights Reserved.

[Twitter](#) [LinkedIn](#)