uStella Rübsaam

GD2B Mediacollege Amsterdam

Februari 2020

**LOGBOEK SPECIAL INPUT/OUTPUT**

EYE MOTION TRACKING

Onderzoek

Ik heb eerst gezocht op eyetracking en eye motion tracking in google. Ik heb natuurlijk even erover gelezen op Wikipedia, ook vond ik op een pysource pagina een tutorial met Opencv in Python. Toen ik ging zoeken op Python, vond ik PyGaze, een open-source toolbox voor eye tracking in Python. Hier vond ik veel blogs van mensen die ook vergelijkbare software aan het ontwikkelen zijn met Python en PyGaze. Hierdoor heb ik besloten mijn programma in Python te schrijven.

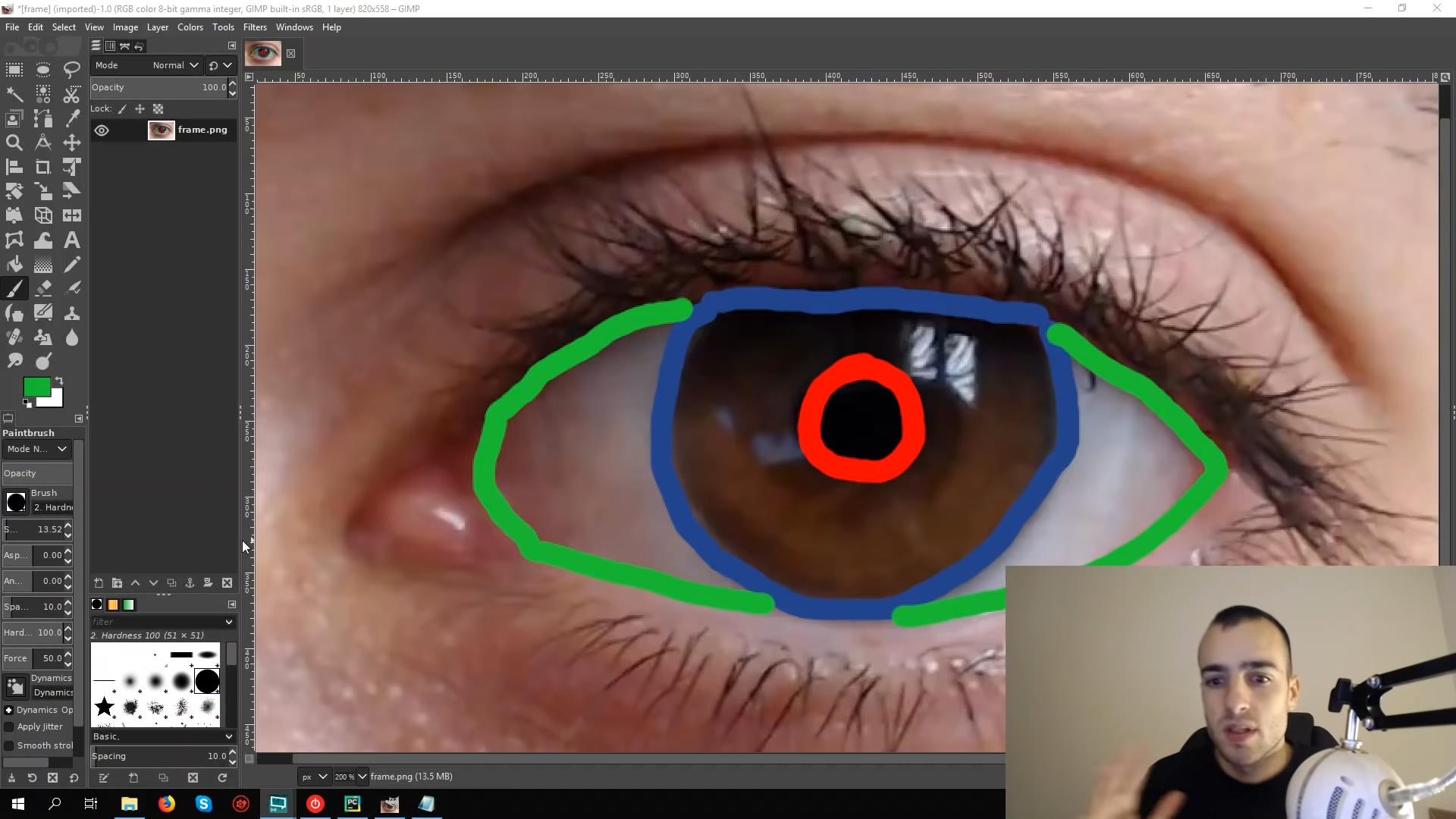
Ontwerp

Ik wil een programma maken wat bijhoudt waar je heen kijkt op het scherm, misschien aangegeven met een bolletje, waar je precies kijkt op het scherm. Met behulp van de webcam, zoekt het programma eerst jouw gezicht, vervolgens je ogen en kijkt dan naar verhouding oogwit en iris, om daaruit op te maken waar je precies heen kijkt. Voor de eventuele game, zou ik gaan voor een SCP setting, waar SCP-173 en SCP-096 in zitten. SCP-173 kan je niet uit het oog verliezen en SCP-096 kan je juist niet aankijken. Ze zullen beiden verschijnen op bepaalde momenten, 173 moet je volgen met je ogen en 096 juist vermijden met je ogen.

Voorbereiding

Na het kort kijken van de tutorial merkte ik al dat ik er niet genoeg van begrijp, omdat ik nooit eerder met Python heb gewerkt. Daarom besloot ik eerst een Python course te volgen.

Voordat ik begon met het programmeren van mijn programma heb ik een virtual environment gemaakt binnen de directory van mijn project zodat ik alle python libraries daar kan downloaden die ik nodig zou hebben.



Struggle blog

**10-2-2020**:

Begin van onderzoek naar eye motion tracking, veel moeite met het vinden van beginner friendly en understandable tutorials of websites.

**24-2-2020**:

Begin Python cursus. Het is erg langdradig aangezien ik dingen als loops en if-statements al veel heb gebruikt in andere talen.

**2-3-2020**:

Gedoe met het installeren van PyGaze packages en libraries, na uren eindelijk virtual environment met libraries werkende gekregen.

**5-3-2020**:

Documentatie huidige PyGaze komt niet overeen met de versie van de package die op de download pagina staat waardoor het testen van de libraries niet kan.

**6-3-2020**:

Bezig met het veranderen van versies van verschillende packages zoals PsychoPy en Pyglet. Daarnaast ook 32 bit versie van Python geprobeerd, aangezien dat misschien beter zou werken met PyGaze. Besloten PyGaze op te geven en naar andere opties te kijken voor dit project.

Bronnenlijst

<https://en.wikipedia.org/wiki/Eye_tracking>

<https://pysource.com/2019/01/04/eye-motion-tracking-opencv-with-python/>

[http://www.pygaze.org/](http://www.pygaze.org/%20)

<https://www.packtpub.com/application-development/application-development-python-and-flask-video>

GRAPHIC TABLET

Onderzoek

Ik heb een tekentablet van Wacom, dus zocht ik naar de specifieke versie (CTL-490/W) en de SDK daarvan, de Common Device Library, Wacom WILL SDK. Aangezien de samples van de SDK in C# zijn gemaakt en voor Visual Studio is bedoeld, heb ik besloten dit project te maken met C# in Visual Studio 2017. Hierdoor heb ik wel uit moeten zoeken hoe je C# files runt en wat je daar allemaal voor nodig hebt (.NET core SDK).

Ontwerp

Ik wil een programma maken dat gebruik maakt van de input van een graphic tablet, met name de druk van de pen, dus hoe hard je met de pen op het tablet drukt. Daarbij dacht ik een game te maken die op het moment dat je het tablet raakt met de pen, een rondje maakt en hoe harder je drukt, hoe groter het rondje wordt. Op het scherm zullen lege rondjes verschijnen die je moet vullen met jouw pen.

Voorbereiding

Voordat ik begon met coderen, heb ik een korte C# tutorial gevolgd zodat ik kon leren hoe je C# files runt en hoe het werkt met SDK’s. Ik heb de goede versie van .NET Framework moeten downloaden, ook drivers voor de tablet moesten nog geïnstalleerd worden.

Struggle Blog

**9-3-2020**:

Informatie vinden over het programmeren met een Wacom tablet was moeilijk te vinden. Er zijn heel veel verschillende SDK’s beschikbaar voor veel verschillende doeleindes. Het runnen van C# files bleek ingewikkelder dan bijvoorbeeld python.

Sample gevonden van Wacom, die de tablet zou moeten detecten, dat werkt alleen niet. Verschillende versies van Visual Studio geprobeerd en een andere sample gedownload.

**11-3-2020**:

Voor het gebruiken van de Wacom en Ink SDK moet je een license generaten, wat nogal veel gedoe is en ik niet van plan ben mijn tijd in te steken.

Bronnenlijst

[https://developer.wacom.com/en-us/developer-dashboard/downloads#wacom-device-kit](https://developer.wacom.com/en-us/developer-dashboard/downloads%23wacom-device-kit)

<https://docs.microsoft.com/en-us/dotnet/csharp/language-reference/compiler-options/command-line-building-with-csc-exe>

<https://stackoverflow.com/questions/271438/how-to-execute-c-sharp-file>

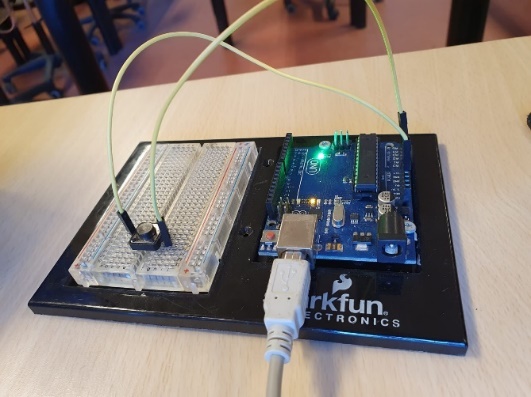
<https://github.com/Wacom-Developer/will-sdk-for-devices-windows-uwp>



ARDUINO

Onderzoek

Voor de Arduino heb ik niet veel onderzocht. Ik heb er gewoon één van school geleend en ik heb daar samen met Armando een beetje mee rondgespeeld. Het Arduino programma, wat gebruik maakt van een versimpelde versie van C++, heb ik natuurlijk gedownload en vervolgens hebben we een klein lampje laten branden d.m.v. een knopje om het te testen en daarna de input in plaats van naar het lampje, naar mijn laptop laten gaan.



Ontwerp

Ik wil Pop the Lock maken met een Arduino. De input is één knop die je op het goede moment moet indrukken. Pop the lock is een spel waarin een balkje rondjes draait en elke keer dat hij op het rondje land, dat ergens op die route staat, moet je op het goede moment op de knop drukken en dan gaat het balkje de andere kant op draaien.



Unity code

Voor het maken van de game heb ik voor bepaalde game mechanics een tutorial gevolgd die functies gebruikte die nodig waren voor deze game die ik nog niet kende. Ook heb ik assets voor de game van deze github afgehaald. Veel heb ik niet van zijn code gebruikt, het ging vooral om de RotateAround die gebruikt moet worden om de paddle rond te laten draaien en een systeem om de gele dots random te laten respawnen (ik had geen idee hoe ik dat zou moeten doen).

User Tests

[User 1 (link)](https://youtu.be/0RRSK7rX2GE):

Uitleg bij het spel was niet nodig, ze drukte gelijk op het knopje. Wel wilde ze er snel mee ophouden omdat het te moeilijk was om de timing goed te krijgen.

User 2:

Hij heeft het spel eerder gespeeld en moest alleen even wennen aan de timing die iets anders was. Wilde graag meer spelen en probeerde de jackpot te halen, dat bleek moeilijk en is niet gelukt. Wel vond hij dat het leuk was en niet te moeilijk, het was namelijk niet de bedoeling dat de jackpot makkelijk is.

Struggle Blog

**12-3-2020**:

Door een rede die ik nog steeds niet begrijp, gaf de Arduino bij geen vol elektriciteitscircuit alsnog lagere volt waardes aan, maar bij het indrukken van de knop gaf hij wel de max waarde aan. Daarom heb ik ervoor gezorgd dat ‘button’ true is als de volt waarde maximaal is.

Bij het maken van de button voor de game liepen we tegen het probleem aan dat de button alleen getriggerd moet worden het moment dat je hem indrukt, niet zolang je hem ingedrukt houdt. Daarvoor moest ik dus handmatig code schrijven, aangezien er geen KeyDown of iets dergelijks is voor Arduino.

**13-3-2020**:

Gebruik gemaakt van een switch case in de Arduino code, aangezien dat beter werkt een veel minder code is. Ik heb een state machine gemaakt, die in de gaten houdt of de knop al eerder is ingedrukt of niet.

**9-4-2020**:

Nieuwe Arduino aan de praat gekregen met de code die ik al had. Heb het circuit een beetje moeten aanpassen zodat hij weer werkte en aan het begin flikkerde het lampje ook als ik de knop niet indrukte, maar dat is nu ook gefixt.

Ik had een probleem met hoe scherp mijn assets waren ingame doordat ik per ongeluk de scale had veranderd. Er voor zorgen dat als je een dot had geraakt, dat dan de volgende verscheen was lastig doordat het random alleen in een bepaalde ring mocht spawnen. Zoiets heb ik nog nooit eerder gedaan.

Ik moest om de System.IO.Ports van Unity te gebruiken, .Net2.0 veranderen naar .Net4.x waar ik pas na een paar google searches achter kwam.

Het bleek nog lastig om een handig stukje code te schrijven in Unity die de onButtonClick registreerd die ik in het Arduino script heb gemaakt.

**10-4-2020**:

Ik heb het elektriciteitscircuit van de Arduino verbeterd door een resistor te gebruiken de de volt value nog specifieker maakt die ik binnen krijg. De volt waardes die zo inconsistent waren, kwamen doordat ik geen resistor gebruikte. Als je dat niet doet en geen vol circuit hebt wanneer de knop niet is ingedrukt, gaan de waardes “floaten”, ze geven willekeurige waardes mee. Ook ben ik in plaats van een analog pin een digital pin gaan gebruiken, aangezien die sneller leest (maar wel minder specifiek).

In Unity heb ik een try catch voor het eerst gebruikt om daarmee de bytes te lezen die de Arduino stuurt. De game werkte nadat ik alle if statements had veranderd. Wel was er een beetje lag wat snel gefixt kon worden door de delay en ReadTimeout zo te tweaken dat er zo min mogelijk lag was.

Gameplay demo

<https://youtu.be/VJX0Qbu7Eio>

Github

<https://github.com/stellarubsaam/SpecialInputOutput>

Bronnenlijst

<https://www.arduino.cc/>

<https://youtu.be/JDN-YlQlYKE>

<https://github.com/indienuggets/Pop-The-Lock-Clone>

<https://www.youtube.com/watch?v=of_oLAvWfSI>