# KiteX Project Overview

Quick Status: On September 13th I succeeded in creating a autonomous flying kite. Source code and components list for the AK002 prototype further down this page!

Why is Kite Energy interesting?

Traditional wind mills is often the preferred solutions when creating new power plants due their low price per watt and environmental impact. **Using kites as a mean of capturing wind energy has the potential to significantly reduce the cost of energy.**

The energy that you can extract from the wind is largely related with the swept area of the wind turbine. A kite is able to sweep a given area with a much smaller system than a traditional turbine. A tether to the ground can replaces most of the *previously* necessary support structure, the inner parts of the wings, the tower and foundation. Makani, the currently leading commercial kite energy project, estimates that their kite system can deliver approximately **10 times the energy per weight** of a traditional wind turbine.
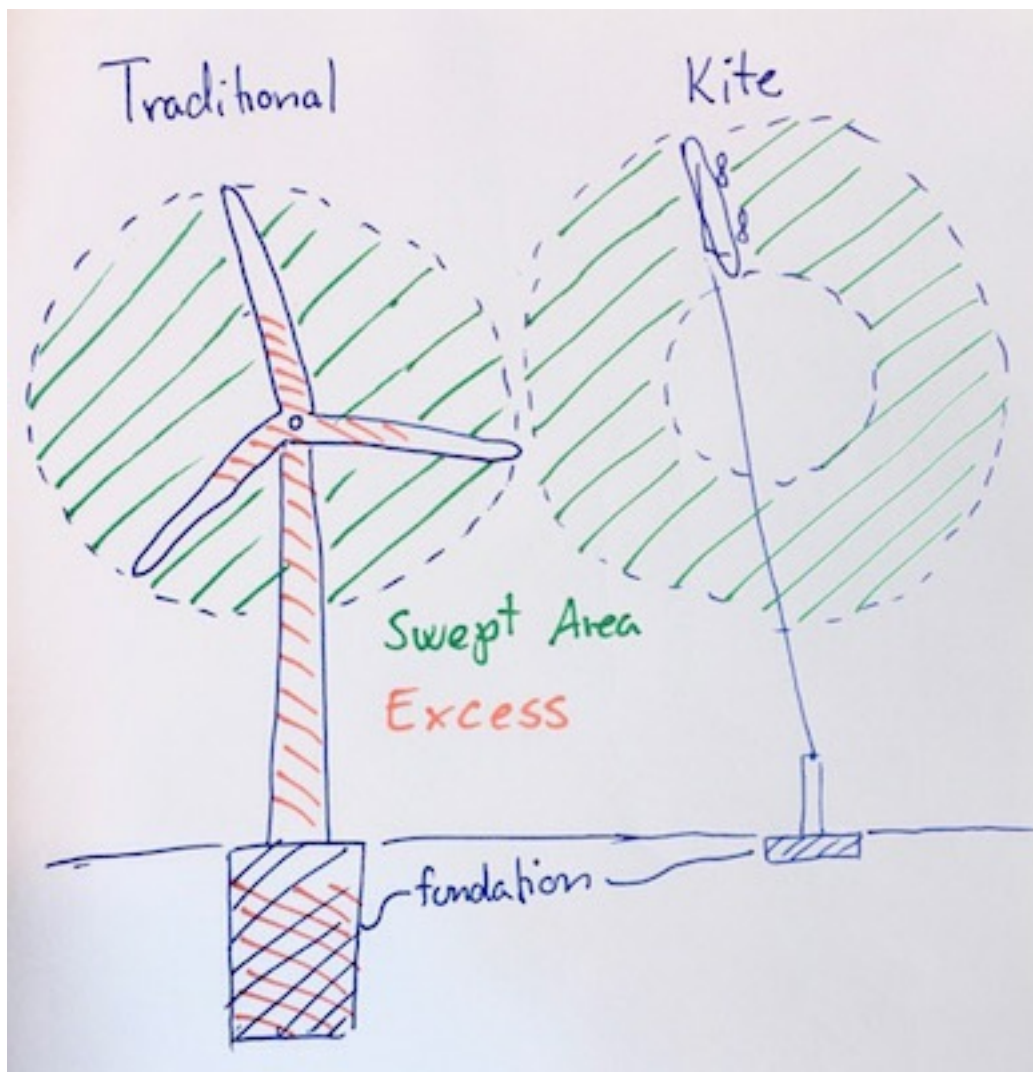
Two further advantages of the kite system is that it can more easily capture wind at higher altitudes where the wind is more consistent, delivering the rated output a higher degree of the time. This will become especially important as a larger portion of the locally generated energy is from "unreliable" sources. Since the bending moment on the foundation of a kite mill is small to non-existent it will be much easier to setup these kite system in offshore conditions.

From a technical perspective it would have been possible to make such a kite ever since the invention of the computer, but right now we are getting to a point in time where advances in materials and computer technology make is a viable and cheaper options.

What is the purpose of KiteX

We would like to promote and carry out the development of kite technology in order to create a future with cheaper and only renewable energy. Cheaper energy will help societies reduce the average amount of "hard" work required to sustain a given living standard freeing up time to pursue more interesting exercises. We find the reflections in these answers quite interesting: Quora – if energy was free (https://www.quora.com/If-energy-was-free-and-renewable-how-would-the-economy-be-fundamentally-different-from-what-it-is-now)

In order to promote this future KiteX will be a project and company focusing on research and development of kite-systems and we will make sure to share all learnings we can, while still retaining necessary knowhow in-house in order to be a viable business.

Who are We?

Andreas Bruun Okholm

I have a mixed design, mechanical and aerospace degree from the Technical University of Denmark. Doing my master degree I created a smartphone based wind measuring tool, which provide value adding functionality to traditional wind meters and most importantly allows kite-surfers to share ultra local wind information. I worked on this project (Vaavud) for 3 years varying between hardware and software development. We managed to create the most sold smartphone wind meter and the first one to measure wind direction in the world.

I have for a long time been fascinated by the power of kites, and been using them for surfing for the past 10+ years. Being able to work with kites is really fascinating and I find that it allows me to use most of my toolset: mechanics, aerodynamics, programming, startups and most importantly it gives a feeling of making a positive impact.

Heading in the same direction?

Do you share the same vision for a the kite future? get in contact! :)

What are the steps?

The plan towards kite energy is designed with the philosophy of dealing with the most difficult problems first – a complete autonomous self-starting and landing kite. Step 1, 2 are just progressions towards this. (subject to change).

1. Create an autonomous flying kite (DONE)
2. Create a tethered drone that can take off and land autonomously
3. Combine 1 and 2 into a completely autonomous kite

4. Add energy generation
5. Make it big

Step 1 – Autonomously flying kite

Information about the Autonomous Kite 001 prototype will follow later.

Autonomous Kite 002 [Video clip (https://youtu.be/O_YaRTxpii8)](https://youtu.be/O_YaRTxpii8)

On September the 13th, 2016 I (Andreas Okholm) managed to carry out the first successful autonomous flight – Check out the video link above!

Overall the system consists of a kite, a motorized kite steering system, a laptop for control and a smartphone for video tracking.
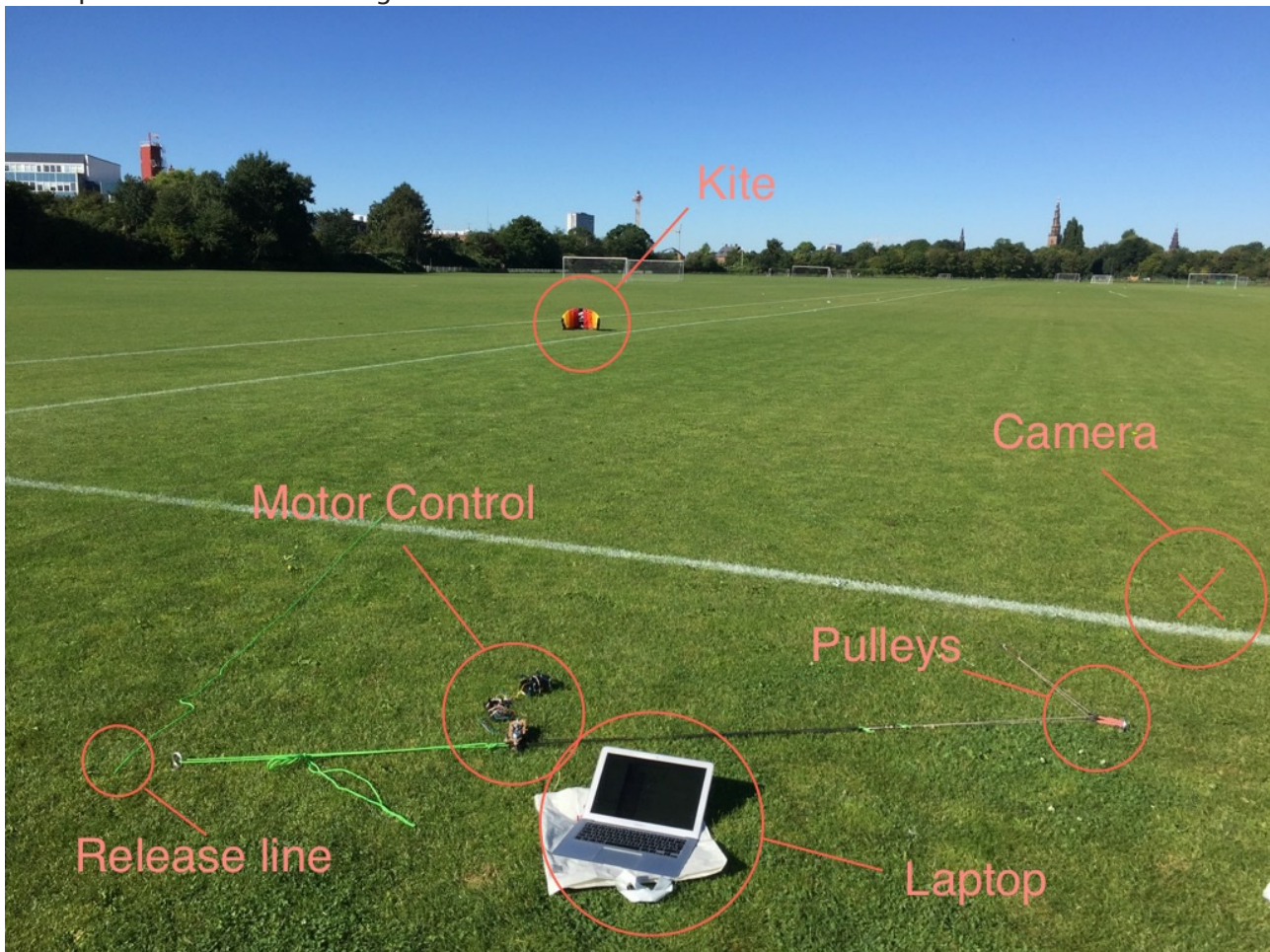
Significant mechanical parts:

- The kite is two lined foil kite HQ Symphony PRO 1.3. The kite were chosen for being the smallest high quality foil kite available. Being so small meant the pull would be the least powerful giving the stepper motor the best chance for staying in control. Thanks to [Dragestedet – Symphony (http://dragestedet.dk/product.asp?product=4179)](http://dragestedet.dk/product.asp?product=4179) for aiding in selecting the best kite. Chose the rainbow color options for best standout against the sky and grass.
- Ronstan Series 20 Orbit blocks. Weighting only 9 grams each these 250 kg (working load) pulleys are light enough not to fall down and effecting the control in light wind and strong enough for the most serious gust.

- NEMA-17 42BYGHW811 Stepper motor. This seems to be one of the most powerful (High torque and high speeds) NEMA-17 Steppers available. High Rated Current, Low Resistance and phase Inductance! ArduinoTech.dk stepper (https://arduinotech.dk/shop/stepper-motor-nema17-42byghw811/) sells this motor incredible cheap seems to deliver reliably.
- Solid storm anchors with the eye close to the ground. While this isn't a specific model I have been very happy with the performance of these *camping* storm anchors Campingshop.dk (http://www.campingshop.dk/stormanker-med-oeje.aspx)
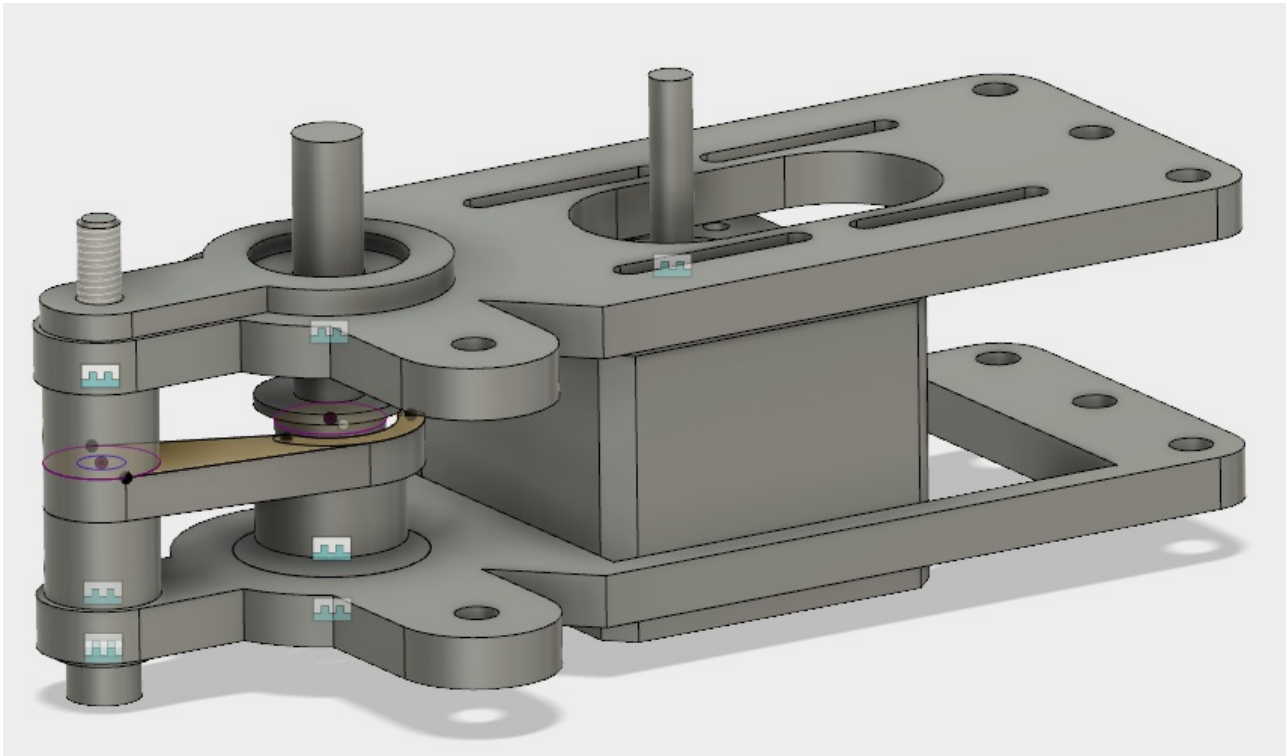
Write me if you want to get some of the Fusion 360 CAD files.

Significant electronics parts:

- EPS8266 Adafruit Huzzah – A full featured easy to use WIFI microcontroller. Flikto.de (https://www.flikto.de/products/adafruit-huzzah-esp8266-breakout) Everything has been working smoothly Once I figured out that I couldn't power it reliably from my Laptop USB Power source and instead used a LiPo battery while being connected with a serial connection with a FTDI cable.
- Arduino Pro Mini 3V 8Mhz. A left over component I had – I would choose a more powerful microcontroller for the stepper motor control next time. Like a Teensy 3.2 or faster.
- DRV8825 Stepper motor driver. These drivers are cheap and can handle up to 2.2 Amps and 45 Volt, which is a lot for the small package and price.
- 120 W DC Boost Converter ebay.co.uk (http://www.ebay.co.uk/itm/272121657453) Perhaps a bit overkill for this application (in terms of current), but it allowed me to get a 38 Volt stepper drive current cheaply.
- Batteries – I have been using a 14.4 Volt 1.5 Ah Lithium-ion battery from a Bosch powertool as the main Motor supply. Could be replaced by a 4 cell lithium-polymer battery. For the logic electronics I have been using a 2000 mAh (ish) 3.7 V single cell lithium-polymer battery from a Power bank.

I also used my Macbook Air and a iPhone 6, but really any laptop and somewhat powerful smartphone will do.

Data flow

The camera observes the position of the kite and estimates the coordinate that best the describes the current position of the kite. This data is sent over Websocket connection over WIFI to the laptop. In a browser windows the tracking data can be observed in realtime and the kite position can be extrapolated in case of shorter tracking problems. The required control input is calculated and send off the ESP8266 microcontroller (also through a websocket over WIFI). From here the information goes to the Arduino which is solely responsible for controlling the step timing of the stepper motor. See the information flow diagram below
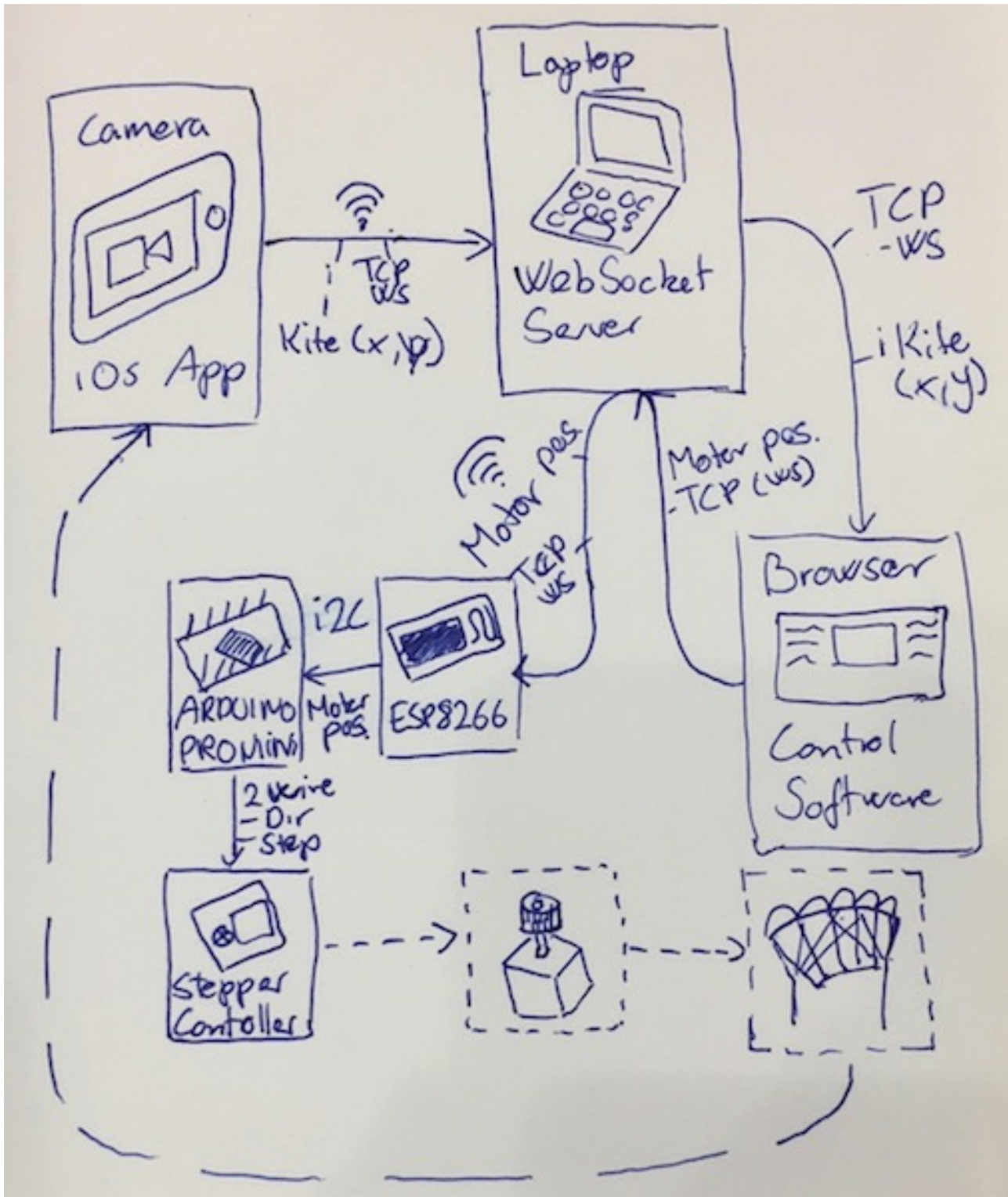
Source code and more can be found under the specific projects

- [ESP8266 Websocket and i2c master code (https://github.com/aokholm/KiteXEspWsSlave)](https://github.com/aokholm/KiteXEspWsSlave)
- [Arduino Pro Mini i2c slave and stepper controller (https://github.com/aokholm/KiteXFastStepper)](https://github.com/aokholm/KiteXFastStepper) I simplified the AccelStepper library and optimized it this use case. Large performance gains by precomputing the acceleration profile. Up from 2000 to 8000 steps per second.
- [iOS GPU Accelerated Kite Tracking app (https://github.com/aokholm/KiteXKiteVideoTracking)](https://github.com/aokholm/KiteXKiteVideoTracking) Thanks to Brad Larson for providing this wonderful framework for doing video manipulation [GPUImage2 (https://github.com/BradLarson/GPUImage2)](https://github.com/BradLarson/GPUImage2)
- [Mission Control – Node.js websocket server and web interface and path following algorithms (https://github.com/aokholm/KiteXMissionControl)](https://github.com/aokholm/KiteXMissionControl)
- [Kite AI Neural Netowrk – not in use (https://github.com/aokholm/KiteXAI)](https://github.com/aokholm/KiteXAI) Using a neural network to the control kite. It does work in the simulator, but currently not in use as I opted for using a pure pursuit path following algorithm instead.

Learnings

A bit of unorganized learnings leading up to the AK002 prototype.

Accces Point

Hack your mac to be an access point even without an internet connection to share. I successful followed this [tutorial (http://www.laszlopusztai.net/2016/02/14/creating-a-wi-fi-access-point-on-os-x/)](http://www.laszlopusztai.net/2016/02/14/creating-a-wi-fi-access-point-on-os-x/) and have been able to make a AP easily every time. Would of cause be nicer it it could be turned on and off from the command line so I could easily do it in a startup script in node

Video Tracking – Open CV, GPUImage

Initially I tried to install and run Open CV on my iPhone. After trying for hours I began seeking other options. I found a [post/article])http://www.sunsetlakesoftware.com/2010/10/22/gpu-accelerated-video-processing-mac-and-ios) by the creator of GPUImage showing GPU accelerated image/color tracking – and though it was perfect. I initially implemented the current solution with the original [GPUImage (https://github.com/BradLarson/GPUImage)](https://github.com/BradLarson/GPUImage), but later realized that a swift version [GPUImage2 (https://github.com/BradLarson/GPUImage2)](https://github.com/BradLarson/GPUImage2) had been made.
The API had changed slightly so I ended up rewriting most of it, with the logic mostly staying the same. GPUImage2 is really easy to work with and very powerful! Great job.

HTTP vs Websocket

I always start out with the simplest solution that I expect to work. In order to control the stepper motor I would send HTTP requests from my iPhone directly to the ESP8266, which also were functioning as an accesspoint. However when getting close to 10-20 request per seconds the ESP would eventually crash. Further the latency were rather high, typical around 100 ms. Eventually I decided to use a websocket instead. It was surprisingly simple to setup and it has been performing great. I were able to achieve a roundtrip latency of around 5 ms and it's rock stable. Digging a little deeper it seems that it would be benificial to have a UDP datalink for the low latency information compared to a TCP which is the underlying protocol of the websocket. However TCP seems good enough for now.

WebsocketServer on Laptop or the ESP

Initially I tried to run the WebSocketServer on the ESP, but I ended up moving the server to the MacBook.

This means I don't have to worry about the WebSocketServer performance. You can do faster iterations and easier debugging.

The server could be written in a high level programming language.

Since all data passes through my laptop it's easy to log it to a file and should be easy to implement the auto guidance system.

It seems like the MB Air has better wifi performance as a router than the ESP8266, meaning longer range for other systems.

Of cause it adds and extra step between the camera and motor controller, but it seems to be worth it.

Artificial Intilligence – Python vs Node

Being a newbie to the field of Machine learning I can highly recommend these two series on machinelearnig:

- Machine Learning Course – CS 156 by Caltech: https://www.youtube.com/watch?v=mbyG85GZ0PI&list=PLD63A284B7615313A,
- Networks Demystified by Welch Labs: https://www.youtube.com/watch?v=bxe2T-V8XRs&list=PLK1aXp7PZ2tMrm1h_JDBcJDhIpbQZPOr1

I used at least a day trying to figure out how to run any deep neural network code in python and never actually managed to run anything if I also wanted to be able to directly plot the results with Matplotlib. I think I tried: scikit-learn, PyBrain and scikit-neuralnetwork in all combinations with System (Python 2.7 installation), PyEnv (3.4, 3.5) and Anaconda. And in every instance there were something wrong. Just wanting to get something running I ended up using Synaptic: http://synaptic.juancazala.com/, which can be run on Node or in your browser – plotting solved! And it worked out of the box! ah. It's sad that Python version and package management can be such a pain.

Cool Articles about Airborne wind energy

- M.L. LOYD. "Crosswind kite power (for large-scale wind power production)" Journal of Energy, Vol. 4, No. 3 (1980), pp. 106–111.
  link (http://edge.rit.edu/content/R15901/public/Matt%20Kennedy/homes.esat.kuleuven.be_~h content_uploads_2011_07_Loyd1980.pdf))
- Diehl, Moritz. "Airborne wind energy: Basic concepts and physical foundations." Airborne Wind Energy. Springer Berlin Heidelberg, 2013. 3–22. link (http://homes.esat.kuleuven.be/~highwind/wp-content/uploads/2013/08/Diehl2013a.pdf)