StayAlive Insektencounter

Schlussbericht

Zuwendungsempfänger: Nicola Wettmarshausen

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums fürBildung und Forschung unter dem Förderkennzeichen 01IS19S08 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation:

Was war Deine Motivation? Welches Problem wolltest Du mit Deinem Projekt lösen?

2018 habe ich die Initiative Ecocurious gegründet – eine Citizen Science Community, die sich mit Projekten an der Schnittstelle Natur, Umwelt und Technik beschäftigt. In diesem Teil-Projekt möchten wir etwas gegen den Biodiversitätsverlust vor der eigenen Haustür, also konkret gegen das Insekten-sterben tun, denn dieses Thema wird aktuell immer brisanter.

Während meiner Recherchen lernte ich, dass das Monitoring von Insekten in der Praxis sehr analog betrieben wird. Fliegende Insekten werden in zeltartigen Malaise-Fallen gefangen und in Alkohol konserviert, um anschließend ihre Biomasse zu bestimmen. Einzelne Familien (Schwebfliegen, Wild-bienen) werden dabei manuell aussortiert und ihre Art bestimmt. So arbeiten beispielsweise die Entomologen im Naturkundemuseum Stuttgart im Auftrag der Landesanstalt für Umwelt BW.

Das brachte uns auf die Idee, die Insekten digital zu zählen und zu klassifizieren, sie dabei aber nicht zu töten. Im Juni 2019 überlegten wir gemeinsam, wie wir eine digitale Echtzeitmessung ermöglichen und dadurch zeigen können, welche Klassen von Insekten (Fliegen, Hummeln, Falter usw.) es noch gibt, wie sich der Bestand über die Zeit verändert und welche Schutzmaßnahmen einen direkten Einfluss auf die Populationszahlen haben können. Wir formulierten das Ziel, ein Citizen-Science-Monitoringtool für Umweltschützer, NGOs, Landwirte, Hobbygärtner, Wissenschaftler und weitere Zielgruppen zu ent-wickeln – ein kostengünstiges Do-it-yourself (DIY)-Gerät auf Raspberry-Pi-Basis, das sich jeder in gemeinsamen Workshops zusammenbauen können soll.

Die Vision: Das Device soll Insekten unterscheiden und lernt dank Künstlicher Intelligenz (KI) ständig hinzu. Die Daten werden später über eine Map im Browser visualisiert. Standorte können verglichen und Tag- und Nachaktivitäten beobachtet werden. So kann jeder selbst zum Bürger-Forscher werden und Nachforschungen anstellen, gezielte Maßnahmen ergreifen und direkt nachvollziehen, ob seine Maß-nahmen wirken. Etwa, wenn er bestimmte nektar- und pollenhaltige Pflanzen in seinem Garten säht statt purem englischen Rasen.

Ausführliche Darstellung der Ergebnisse

Wie war die geplante Vorgehensweise zur Problemlösung (auch Angabe der wichtigsten Meilensteine)? Welche konkreten Ergebnisse hast Du erzielt? Konnten alle Meilensteine erreicht werden?

Im Rahmen des Software Sprints konnten wir einen ersten funktionsfähigen Prototyp (Hardware und Software) für beide Messmethoden (optoakustisch und akustisch) entwickeln, einschließlich dem dazugehörigen Datenhandling für beide Messmethoden (*Insects Monitoring Lab* als Prototyp für die Bilderkennung und eine prototypische Plattform für die Wingbeats-Daten).

Da die konkrete Umsetzung der Idee eines Insektencounters eine parallele, teilweise voneinander abhängige Hardware- und Softwareentwicklung erfordert und deshalb nicht innerhalb eines halben Jahres komplett zu realisieren ist, mussten wir unseren experimentellen Ansatz immer wieder anpassen und iterativ umsetzen. Unser Ziel war es, zwei Messmethoden auszuprobieren und einen Proof-of-Concept zu entwickeln. Wir starteten mit diesen zwei Ansätzen parallel in zwei Teams.

Team Eins in Stuttgart/Tübingen (Jürgen, Matthias, Nicola) widmete sich der Anwendung der *Wingbeats*-Messmethode, zu der es bereits Forschung im Bereich Schädlingsbekämpfung gibt. Wir wollten diese Methode für unsere Fragestellung anpassen: Schaffen wir es, möglichst viele fliegende Insekten (und nicht nur „Schädlinge“ wie Moskitos oder Olivenfliegen) damit zu detektieren? Und welche Hardware brauchen wir dazu? Da das Device später von Usern aller Altersstufen benutzt werden soll, testeten wir keine Laser, sondern Infrarot-LEDs. Diese LEDs wurden mit Photodioden auf der gegenüberliegenden Seite eines 3D-gedruckten Rahmens kombiniert (siehe Skizze). Fliegt ein Insekt durch den Rahmen, schattet sein Flügelschlag das IR-Licht im Frequenzbereich von 15-1000 Hz ab. Diese Abschattung kann spezifisch für eine bestimmte Art oder zumindest Familie von Insekten sein. Das Bestechende an dieser Messmethode ist, dass man später ein Audiosignal verarbeitet (Wave-Format), das man zuerst optisch misst und anschließend wandelt. Auf diese Weise stören zum Einen akustische Ereignisse wie Hundegebell, Autos, Flugzeuge etc. nicht und zum Anderen können optische Störungen leichter herausgefiltert werden.

Wir arbeiteten hier mit der Software GNU Radio (https://www.gnuradio.org/about/) und übertrugen die Filtereinstellungen später in Python (siehe Github). Mit GNU Radio entstörten Matthias und Jürgen das ankommende Licht-Signal, modulierten und demodulierten es und testeten verschiedene Filtermöglich-keiten. Als wir dann im Spätherbst so weit waren, Wespen von Stubenfliegen unterscheiden zu können, begann allerdings die Winterperiode. Gekaufte Insekten waren nicht in dem Umfang vorhanden, wie wir es uns erhofft hatten. Hinzu kam, dass Insekten wie Florfliegen, Wachsmotten, Heimchen oder Wüstenheuschrecken partout nicht fliegen wollten und dadurch mit dieser Methode nicht messbar waren. Dadurch reichte die Datenmenge nicht aus, um ein KI-Modell trainieren zu können. Wir fingen dennoch an, ein vortrainiertes Modell mit Trainingsdaten von 6 Mückenarten (Daten von iNaturalist aus https://www.kaggle.com), bestehend aus einem 1D-CNN (eindimensionales Neuronales Netz) mit Daten von bewegungsfreudigeren Fleischfliegen (Sarcophagidae) mithilfe von TensorFlow (TF) weiter zu trainieren. Dabei nutzten wir TF, um später mit der abgespeckten Version TF Lite (8 Bit) auf dem Raspberry weiterarbeiten zu können.

Auf der Scicar-Konferenz 2019 (https://www.scicar.de/) knüpfte ich Kontakte zur Uni Saarbrücken und konnte ein studentisches Team (fünf Studierende) für unser Projekt gewinnen. Matthias entwickelte mit mir gemeinsam drei Arbeitspakete für den Zeitraum November bis Ende Januar, die die Studierenden im Rahmen ihres Software-Engineering-Projekts (https://cms.sic.saarland/se/) für Ecocurious umsetzen

konnten. Er plante das Backend, definierte die Schnittstellen, wählte die Tools und Bibliotheken aus und kontrollierte den Arbeitsprozess des Saarbrücker Teams. Wir definierten zahlreiche Software-Anforderungen für die drei Tasks Sensor/Raspberry Pi, Server und Webseite, die vom Team größtenteils auch so umgesetzt wurden, beispielsweise:

- Trigger: Im Sensor haben wir eine akustische Detektionsschwelle eingerichtet, die definieren soll, ob ein Signal als Flügelschlag wahrgenommen und daraufhin aufgezeichnet wird. Das Gerät soll nur dann aufzeichnen, wenn ein Insekt durch den Sensor fliegt, um möglichst wenig Datenmüll zu generieren, muss dann aber sofort reagieren. Da das Signal sehr kurzlebig ist (200-800 ms), reichen kleine Verspätungen aus, um ein Insekt nicht zu detektieren.
- Der Sensor soll Daten über LoraWAN übertragen können (nur Zahlen, keine Bilder) und unterstützt MQTT optional zu HTTP.
- Lightweight: Da die Audioaufnahme, Audioverarbeitung als auch die Klassifizierung der Insekten auf einem Gerät unter limitierten Hardware-Ressourcen eines Einplatinencomputers laufen muss, müssen ressourcenintensive Prozesse wie Datenspeicherung, Modellgenerierung und Weiterverarbeitung ausgelagert werden. Die WAV-Dateien werden deshalb auf eine SD-Karte geschrieben, die regelmäßig ausgetauscht wird. Steht eine Internetverbindung zur Verfügung, werden die Dateien auf einen Server geschrieben und dauerhaft archiviert. Dies haben wir so umgesetzt.
- Schreibzyklen: Da der Speicher des Raspberry Pi's über eine limitierte Anzahl an Schreibzyklen verfügt, speichern wir temporäre Daten im Normalbetrieb in einer RAM-Disk.
- Das Backend besteht aus mehreren Komponenten (Datenbankserver, SensorThingsAPI, Userschnittstelle zur Verwaltung, Schnittstellen für die Visualisierung). Diese Komponenten liegen in Docker-Containern vor und können per Docker-Compose gestartet werden.
- Webseite: Über den noch einzurichtenden öffentlichen Server soll jeder User seinen Sensor anmelden können. Bisher ist dies nur dem Entwicklerteam zugänglich. Die Userverwaltung soll einfach zu bedienen sein und Datenschutz garantieren (https://insectcounter.zapto.org/). Die browserbasierte Map kann mehrere Sensoren auf einmal visualisieren und komplexere Visualisierungen anbieten (über eigenen JavaScript-Code). Sie hat per SensorThings-API Zugriff auf die Daten.
- TensorFlow (TF): Wir haben das Modelltraining mit TF und die Konvertierung in TF Lite für den Raspberry begonnen.
- Erweiterbarkeit: Da Machine-Learning-Modelle nur so gut sind wie ihre Trainingsdaten, wollen wir das bestehende Modell mit neuen Daten weiterentwickeln können, ohne dass dafür eine Softwareanpassung nötig ist. Im Idealfall sollte das alleinige Hinzufügen der Daten in einen Ordner genügen, um das Modell zu erweitern. Dies wurde implementiert.
- Für die wissenschaftliche Auswertung wurde auf dem Sensor auch die Möglichkeit implementiert, Umgebungsdaten wie Temperatur, Luftfeuchte etc. abzufragen und mit den Ergebnissen der Klassifizierung zu übertragen.
- Sprache: Um die Erweiterbarkeit des Projektes zu vereinfachen, soll die Anzahl der verwendeten Programmiersprachen möglichst gering gehalten werden. Daher ist ein Großteil in Python programmiert worden, auch um Hardwarekompatibilität zu garantieren.

Wichtig war es uns, möglichst viele standardisierte Komponenten zu verwenden. Im Sensor etwa wurde der Fokus auf die Verwendung von bekannten Bibliotheken zur Audiobearbeitung und Klassifizierung gesetzt. Auch im Backend wurde Wert auf Standards gesetzt. Daher fiel die Wahl hier auf GOST, eine Referenzimplementierung der OGC SensorThingsAPI. Diese Schnittstelle erfreut sich wachsender

Verbreitung im Open Data Umfeld. Sie erlaubt es, Daten sowohl zeitlich als auch räumlich abzufragen, beispielsweise, um die Vorkommen einer Insektenart A im Naturschutzgebiets XY mit der Umrandung Z (GeoJSON Objekt) anzuzeigen. Durch die Verwendung von Docker-Containern ist der Nachbau auch für Laien auf einfache Weise möglich.

Die Zusammenarbeit mit den Studierenden verlief rein virtuell über Slack, Mail, Videokonferenz und Telefon. Sie war aufgrund der Komplexität des Projekts, der unterschiedlichen Backgrounds der Studierenden und den Vorgaben der Hochschule sehr zeitintensiv in der Kommunikation.

Derweil arbeitete **Team Zwei in Berlin** (Stefan, Levin, Oskar und Roman für die Hardware) mit der rein optischen Messvariante (fotografisch/Computer Vision). Nachdem Roman ein Terrarium gebaut hatte, zogen dort Heuschrecken und Heimchen ein. Weil es das Ziel des zu entwickelten DIY-Devices ist, mit einer kostengünstige Technik zu arbeiten, nutzten wir eine PiCam zum Fotografieren. Dabei wollten wir sicherstellen, dass die sich bewegenden Insekten in einer Bildfolge nur einmal gezählt werden. Darum verwendeten Stefan und Levin die Open-Source-Überwachungssoftware *Motion* (>https://motion-project.github.io/). Auch hier bauten sie einen Trigger ein: Aufgezeichnet wird nur, wenn ein Insekt sich bewegt (Datensparsamkeit). *Motion* kann auch mit hochwertigeren Kameras arbeiten und ist daher für die spätere Erweiterung des Setups optimal.
Die mit dem Raspberry Pi generierten Bilddaten werden anschließend in einer zentralen Datenbank hochgeladen, die zunächst auf unseren Servern laufen soll (noch gibt es keinen Projektserver). Diese Datenbank muss später an die Struktur der Europäischen Datenbank Fauna Europaea (https://fauna-eu.org/) angepasst werden, um wissenschaftlich nutzbar zu sein.

Da es Winter wurde, mangelte es auch den Berlinern an einheimischem Artenreichtum für ein sinnvolles Modelltraining der KI. Darum haben wir uns entschieden, zuerst mit einem anderen Task zu beginnen: Einer Webplattform fürs gemeinsame Datenmanagement (Arbeitstitel: *Insects Monitoring Lab*, http://195.201.97.57/, https://github.com/ecocurious/insect-webtool). Die Grundidee ist dabei, die Insekten-Bestandsdaten in einer zentralen Datenbank mit der Öffentlichkeit zu teilen. Die an verschiedenen Orten generierten Daten sollen dabei nicht nur von Citizen Scientists, sondern auch von Wissenschaftlern und Naturschützern genutzt werden. Die breite Zielgruppe von technisch versierten bis nicht-versierten Usern ist dabei maßgeblich für die Usability der Plattform.

Für das Team Berlin (das ja eigentlich die KI bearbeiten wollte) bedeutete dies, sich in neue Themenfelder einzuarbeiten. Die Backendentwicklung der Plattform war dabei unproblematisch, da Stefan und Levin ihre Erfahrungen in der Strukturierung von Webshops einbringen konnten. Allerdings mussten sie sich in die Frontendentwicklung erst neu einarbeiten, was die Umsetzung der einzelnen Aufgaben verzögerte. Bis zum *Demo Day* bauten sie eine Plattform mit diesen Funktionalitäten:

- Übertragung von dezentral gesammelten Bildern (Raspberry Pi) in eine zentrale Datenbank
- Suchfunktion für diese Bilder entwickeln
- Erstellung von Datensätzen zum Trainieren von ML-Algorithmen
- Annotation der Bilder (Markierung und Kategorisierung von Insekten)
- Python Client, der den Download von Datensätzen und den Upload von annotierten Daten vereinfacht
- Visuelle Evaluation von maschinell (KI) klassifizierten Bildern
- Darstellung von Zeitreihen (Entwicklung der Insektenpopulation über die Zeit)

Ins Frontend baute des Team Berlin deshalb folgende Funktions-Elemente ein:

Live-View: Um später jeden autonom im Freiland messenden Sensor überwachen zu können, wurde eine Live-View-Ansicht eingebaut (zunächst für nur einen Sensor).

Browser: Der User kann durch die Fotos browsen, sich einen Zeitraum auswählen und die Bilder zur Offline-Bearbeitung als Datenpaket auf seinen Rechner herunterladen. Er kann sie auch vom Raspberry Pi direkt hochladen (Transfer in beide Richtungen). Er kann sich außerdem in einer Visualisierung anzeigen lassen, wieviel Insekten einer Art in den ausgewählten Datenmengen vorkommen.

Annotation: In dieser Ansicht können die Insekten-Arten in den einzelnen Bildern annotiert werden. Die Annotationen sind farblich codiert und damit einem User zugeordnet. Das hat den Vorteil, unseriöse Tagger herausfiltern zu können und so die Datenqualität zu verbessern.

Die Annotationen trainieren das KI-Modell YOLO (You Only Look Once/ https://pjreddie.com/darknet/yolo/), das derzeit auf dem Desktop läuft. Das Team verwendete YOLO, weil hier zwei Aufgaben zugleich erledigt werden mussten: Einmal die Frage, wieviel Insekten auf dem Bild zu sehen sind (=Objekterkennung), und gleichzeitig die Frage, welche Insekten es sind (=Klassifizierung). YOLO ist sehr gut geeignet, diese Fragen gleichermaßen beantworten zu können.

Mit den Heuschreckenarten aus dem Terrarium labelte Levin 400 Bilder und erstellte so erste Trainingsdaten für das KI-Modell. Er stellte fest, dass die Heuschrecken mithilfe von YOLO zuverlässig erkannt und die vom Team Berlin entwickelte Plattform *Insects Monitoring Lab* grundsätzlich funktioniert. Allerdings sind viele Funktionen der Plattform noch nicht voll ausentwickelt. Eingebaut werden soll noch die Unterstützung von mehrschrittigen Annotationsprozessen, die Strukturierung der Label-Hierarchie in der Datenbank wie bei Fauna Europaea, die Nutzbarkeit mehrerer Raspberry PI-Geräte, Optimierung des User Managements und das Management von KI-basierten und gleichzeitig User-basierten Annotationen. Um das *Insects Monitoring Lab* voll funktionsfähig und stabil zu machen, wären weitere Arbeitsschritte im Bereich Frontend und Backend im Umfang von mindestens 300 Arbeitsstunden nötig.

Beitrag des Projektes zu den Zielen der Förderinitative „Software-Sprint“

Wer ist die Zielgruppe für Deine Lösung? Wie profitiert sie von den Ergebnissen Deines Projekts?

Die gesamte Idee, Insekten digital zu zählen und zu klassifizieren ist ein Bottom-up-Citizen-Science Projekt (bürgergetrieben und nicht durch eine Hochschule gestartet). Über die noch zu gestaltende Homepage zum Projekt und insbesondere über die *Insects Monitoring Lab* Plattform wollen wir verschiedene Interessengruppen vernetzen und ihre Zusammenarbeit unterstützen. Dabei handelt es sich um

- Maker, also Entwickler von Hardware-, Software- und IoT-Lösungen. Sie profitieren von unseren entwickelten Tools für den Raspberry PI.

- Data Scientists, die KI zur Klassifikation entwickeln. Sie profitieren von unseren dokumentierten Forschungsergebnissen zum Machine Learning Framework YOLO auf Insekten-Daten

- Biologen können von einem integrierten Daten-Annotations-Tool profitieren, um ihr Wissen mit Kolleg/innen und Bürgern teilen zu können. Sie können außerdem die analysierten Daten zum Insekten-Bestand für ihre Forschung nutzen.
- Bürger können den Insektenbestand auf ihren ausgewählten Flächen messen, sich darüber austauschen und dabei einiges über Forschung, eine umweltbezogene Anwendung der KI und das Insektenverhalten von beobachteten Arten lernen. Sie können außerdem lernen, was passiert, wenn man Insekten im Garten das richtige Futter anbietet: Nektar- und pollenreiche einheimische Blüten beispielsweise.

Welche Bezüge gibt es zu den Themenfeldern und Zielen des Software Sprints?

Unter dem Themenfeld *Civic Tech* ermöglichen wir interessierten Bürgern, ein kostengünstiges technisches Gerät auf Raspberry Pi-Basis aus Bauteilen in Workshops selbst zusammenzubauen, eigene Daten zu Insekten-Beständen aufzuzeichnen und diese mit einer Community zu teilen. Alle Roh-, und aufbereiteten Daten und Analysen stehen der Öffentlichkeit auf der Plattform zum Anschauen, Auswerten und zum Download zur Verfügung.

Welche zusätzlichen Erkenntnisse hast Du aus der Projektarbeit gewonnen, auch im Hinblick auf die Begleitung durch die Open Knowledge Foundation?

*A*ls Projektleiterin habe ich sehr vom angebotenen Coaching profitiert, da es nicht immer einfach ist, mehrere Arbeitsgruppen aus der Ferne zu steuern, ohne bei Problemen mal eben ein physisches Treffen einberufen zu können. Darum lief die Zusammenarbeit im Team Stuttgart/Tübingen am besten, da wir uns hier auch kurzfristig treffen konnten. Es war für mich außerdem sehr wichtig, sich mit den anderen teilnehmenden Projektgruppen im Rocket Chat und telefonisch auszutauschen und Tipps im Hinblick auf Projektsteuerungstools, Konferenzteilnahmen und best-practice zu bekommen.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Welcher Nutzen ergibt sich für die Zielgruppe aus den Ergebnissen Deines Projekts?

Siehe oben

Welche weitergehenden Effekte ergeben sich aus der Open-Source-Stellung der Ergebnisse?

Maker und Entwicklerinnen werden aufmerksam und machen mit. Sie sehen, was fehlt und können ihre Ideen einbringen. Das Projekt ist ein Community-Projekt. Wir brauchen weitere Mitmacher, denn das Projekt soll weiter wachsen.

Gibt es Ideen für die Weiterentwicklung Deiner Lösung und Pläne zu deren Umsetzung?

Wir möchten in Zukunft mit Hochschulen im Bereich IT, KI, IoT und Sensorik zusammenarbeiten, ebenso mit Biologen/Entomologen, Vereinen und mit lokalen Umweltgruppen wie NABU, BUND etc.

Hat die Arbeit in dem Projekt Dich in Deiner persönlichen, fachlichen Weiterentwicklung unterstützt?

Wir alle haben gelernt, wie wir konkrete technische Herausforderungen lösen können. Ein ganz wichtiger Lerneffekt war, die Perspektiven von verschiedenen Nutzern, insbesondere Biologen, zu hören und einzubinden. Projektleitung: ich habe gelernt, dass die Tools im Projektmanagement immer auf die Bedürfnisse aller abgestimmt werden müssen, sonst funktioniert es nicht. Das ist teamspezifisch, oft sogar personenspezifisch und muss angepasst werden. Eine generelle Lösung (jeder muss Tool XY als Haupt-Kommunikationskanal benutzen) gibt es leider nicht.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Gab es Arbeiten bzw. Lösungsansätze, die nicht weiter verfolgt wurden? Was waren die Hintergründe, und wie bist Du alternativ vorgegangen?

In den Wintermonaten gab es kaum einheimische Insekten im Freiland und deshalb fehlten uns KI-Trainingsdaten, um anschließend YOLO zu trainieren. Team Berlin hat daher Fokus auf die Plattform *Insects Monitoring Lab* gelegt, die dann in der kommenden Insektensaison die Entwicklung und Skalierung des Insekten-Monitorings unterstützen kann. Bei der Entwicklung des *Insects Monitoring Lab* gab es für das Berliner Team ebenfalls Sackgassen, beispielsweise die Verwendung von GraphQL als Backend API. Dabei hat sich herausgestellt, dass sie die API nur schwer ins Frontend integrieren können. Daher haben sie auf eine Websocket-basierte Frontend-Backend-Kommunikation umgestellt, und dies klappte gut.

Als Sackgasse entpuppte sich der Versuch, existierende Datensätze von Insekten (iNaturalist-Foto-Daten auf der ML-Competition-Plattform Kaggle (https://www.kaggle.com/) zu verwenden und damit ein KI-Modell zu trainieren. Die Fotos sind perspektivisch zu nah aufgenommen (teilweise Makroaufnahmen) und funktionieren deshalb in unserem Setting (totalere Bildperspektive, andere Bildqualität) nicht. Darum ist der Ansatz, komplett mit eigenen Bildern zu arbeiten, hier der sinnvollere.

Im Team Stuttgart/Tübingen haben wir an der Wingbeats-Signalverarbeitung gearbeitet und mit Unterstützung der Saarbrücker Studierenden den kompletten Daten-Workflow für Wingbeats prototypisch abbilden können. Bereits vorhandene Wingsbeats-Aufnahmen (ebenfalls auf Kaggle) bezogen sich auf Tierarten, die bei uns nicht heimisch sind. Teilweise mussten die Audioformate angepasst (alle auf eine Länge gekürzt etc) werden.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Wo können sich Interessenten detailliert über Deine Projektergebnisse informieren (z.B. Webseite, GitHub, Veröffentlichungen)?

Momentan bei Github und auf der Ecocurious-Homepage (eine neue Seite bzw. Unterseite ist geplant)
https://ecocurious.de/projekte/insect-counts/
https://github.com/ecocurious/insect-webtool
https://github.com/ecocurious/insects-count
https://github.com/ecocurious/insectcounter

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Gab es im Projektverlauf Ereignisse, die eine Anpassung der Planung erforderlich machten – z.B. Mehr- oder Minderaufwand bei der Bearbeitung von Teilaufgaben?

Die Gesamtidee ist so umfangreich, dass wir diese Prototyp-Phase mit einem definierten und vorzeigbaren Zwischenstand abgeschlossen haben.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Gab es Entwicklungen anderer Personen oder Institutionen, die Einfluss auf Deine Arbeiten und die Zielsetzung hatten? Wenn ja, worin bestand dieser und wie bist Du damit umgegangen?

Es gibt es mehrere akademische Studien, die die Wingbeats-Messmethode nutzen und auf denen unser Vorgehen teilweise basiert (Quellen siehe unten). Diese Paper ermutigten uns, mit dieser Messmethode weiterzuarbeiten, da sie einerseits störungsarm und freilandtauglich, andererseits sehr kostengünstig ist, was die Hardware-Komponenten angeht. Die Messmethodik ist in außerdem technischen Laien einfach zu vermitteln und für Citizen-Science-Anwendungen daher sehr geeignet.

Quellen sind u.a.:
https://www.kaggle.com/potamitis/wingbeats/kernels
https://ieeexplore.ieee.org/document/7091854
https://ieeexplore.ieee.org/document/8553542
https://www.hindawi.com/journals/js/2018/3949415/
http://www.cs.ucr.edu/~eamonn/CE/Bayesian%20Classification%20withInsect_examples.pdf
https://academic.oup.com/sysbio/article/68/6/876/5368535