

KI zur autonomen Erkennung von Wölfen

Teilnehmende (mit Alter): Paul Zörb (15)

Erarbeitungsort: Gymnasium Athenaeum Stade

Projektbetreuende: Herr Dr. Carmesin, Frau von Bargen

Fachgebiet: Arbeitswelt

Wettbewerbssparte: Jugend forscht

Bundesland: Niedersachsen

Wettbewerbsjahr: 2025



(Abb. 1)

Kurzfassung

Mein Ziel ist es ein Kamerasystem zu bauen, das mit Hilfe einer KI autonom Wölfe erkennt und einen Verantwortlichen alarmiert. Auf die Idee kam ich, weil gerade in der Region Stade oft Rinder oder Schafe, die wichtig für den Küstenschutz sind, gerissen werden und es noch keine nachhaltige Lösung für dieses Problem gibt, die auch den Tierschutz berücksichtigt. Laut einer Wolfsbeauftragten, mit der ich zusammenarbeite, sei es von entscheidender Bedeutung einen Wolfsriss von einem Hunderiss unterscheiden zu können, da 50% aller vermeintlichen Wolfsrisse von Hunden verursacht sind. Bisher beruhte die Einschätzung auf Begutachtung des toten Tieres. Ein Bildnachweis durch mein Kamerasystem kann durch die Weiterentwicklung zum Vorjahresprojekt nun erbracht werden, da die Software um eine Segmentierung und das Kamerasystem um weitere neue Module verbessert wurde. Das wurde experimentell nachgewiesen. Diese Verbesserungen ermöglichen eine Zuordnung des Tieres und optimieren das Wolfsmonitoring.

Inhaltsverzeichnis

<u>1. EINLEITUNG</u>	4
1.1 IDEE.....	4
1.2 ZIEL	4
<u>2. VORGEHENSWEISE, MATERIALIEN UND METHODEN</u>	4
2.1 VORGEHENSWEISE	4
2.2 SOFTWARE	5
2.2.1 VERWENDETE MODULE.....	5
.....	6
2.2.2 PROGRAMMIERUNG	6
2.3 HARDWARE.....	7
2.3.1 VERWENDETE HARDWARE	7
<u>3 ERGEBNISSE</u>	8
3.1 VERSUCHE.....	8
3.1.1 VERSUCHE MIT KATZEN	9
3.1.2 VERSUCHE MIT HUNDEN	10
3.1.3 VERSUCH MIT HUND UND VIDEOSTREAM	10
3.1.4 VERSUCHE MIT WÖLFEN UND VIDEOSTREAM	12
3.1.4 VERSUCHE MIT KATZE UND FERTIGEM SYSTEM	14
<u>4 ERGEBNISSE UND ZIELERREICHUNG.....</u>	15
4.1 ZIELERREICHUNG	15
4.2 ERGEBNISDISKUSSION	16
4.3 AUSBLICK.....	16
<u>5. QUELLEN</u>	17
5.1 QUELLEN	17
5.2 ABBILDUNGEN	17
5.3 TABELLEN	17
<u>6. DANKSAGUNG</u>	18

1. Einleitung

Heutzutage liest und sieht man immer wieder in den Nachrichten Meldungen von Wölfen, die Tiere wie Schafe oder Rinder reißen. Ein Beispiel dafür ist der Stader-Tageblatt Artikel (2023) „Lauter Ruf nach Wolfsabschüssen“ vom 25.11.2023, der von Grit Klemp geschrieben wurde. In diesem berichtet der Wolfsberater Michael Kohlhof über den aktuellen Stand zum Wolf in der Region Stade. Die Ausbruchssicherheit der Weidetiere und die Wolfsabwehr gleichzeitig durch Zäune zu gewährleisten sei eine Herausforderung, wenn nicht unreal. Nach Kohlhof braucht man verifizierte Beweise, wie Fotos und eine entsprechende Dokumentation. Dies bestätigte auch die Wolfsberaterin des Landkreises Hamburg-Harburg, Svenja Oßenbrügge. Laut ihr wäre ein Kamerasytem sehr hilfreich, um Fotos, bzw. Videos zu bekommen, da man so unter anderem einfacher erkennen kann, ob es sich bei dem Wolfsriss um einen Wolf, oder einen Hund handelt. Tatsächlich wären ca. 50% aller vermeidlichen Wolfsrisse von Hunden. Dies sei jedoch anhand des toten Tieres schwierig festzustellen. Außerdem würde ein solches Kamerasytem das Wolfsmonitoring deutlich vereinfachen ebenso wie das Ermitteln der Wolfsbestände.

1.1 Idee

Ich war schon immer ein Tierfreund und mich hat die Diskussion über die Wölfe als Problem und der Lösung durch Abschuss betroffen gemacht. Daher möchte ich eine KI programmieren, welche automatisch Wölfe erkennt. Anschließend möchte ich diese KI mit einer Kamera verbinden. Damit wäre das von Michael Ohlhof und Svenja Oßenbrügge genannte Problem, dass man Beweise und eine Dokumentation braucht gelöst. Außerdem könnte innerhalb kürzester Zeit der Jäger oder Hirte alarmiert werden, damit dieser den Wolf vertreiben bzw. abschießen kann. Mit dieser Kamera könnten sämtliche Informationen über die Wolfsrudel problemlos dokumentiert werden. Außerdem wäre es nicht mehr nötig, wahllos alle Wölfe in einem Gebiet rund um den Wolfsriss abzuschließen, sondern man könnte dies gezielt auf die Wölfe eingrenzen, die die Tiere gerissen haben.

1.2 Ziel

Zuerst möchte ich das neuronale Netz aus dem vorherigen Jahr in Bezug auf die Genauigkeit der Segmentierung verbessern. Dafür werde ich YOLOv8 verwenden. Das neuronale Netz soll außerdem in der Lage sein, nicht nur Wölfe zu erkennen, sondern auch Hunde, damit eine Unterscheidung des Tieres und somit eine genauere Datenlage zu den Wolfrissen möglich wird.

Anschließend möchte ich das neuronale Netzwerk mit einer funktionalen Kamerabox verbinden, welches automatisch scharfe Fotos schießt, diese analysiert und wenn ein Wolf oder Hund zu sehen ist eine SMS an den Verantwortlichen sendet, damit dieser schneller handeln kann.

2. Vorgehensweise, Materialien und Methoden

2.1 Vorgehensweise

Letztes Jahr hatte ich meine KI mit TensorFlow trainiert. Dieses System hatte jedoch einige Nachteile. Zum einen konnte das Netzwerk nicht zwischen dem Hintergrund und dem eigentlichen Objekt differenzieren. Ein weiteres Problem war, dass

das neuronale Netzwerk nicht gut verallgemeinern konnte. Das heißt es funktionierte gut mit Bildern, die den Trainingsbildern sehr ähnlich waren. Da selbst geschossene Bilder von Tieren jedoch deutlich anders als waren, als die Trainingsbildern, war die KI unter realen Bedingungen deutlich schlechter. Daher bin ich auf YOLOv8 (2.2.2) umgestiegen. YOLOv8 hat eine Hintergrundsegmentierung, das heißt die KI entscheidet erst, welche Objekte im Bild von Relevanz sind und klassifiziert diese anschließend. Das neuronale Netzwerk mit YOLOv8 ist auch deutlich besser im Verallgemeinern der Trainingsbilder. Anschließend habe ich das neuronale Netzwerk so angepasst, dass es sowohl Hunde, als auch Wölfe erkennt, da laut der Wolfberaterin Svenja Oßenbrügge, die Hälfte aller vermeidlichen Wolfrisse von Hunden seien. Daher ist es sehr wichtig, dass meine Kamera sowohl Hunde als auch Wölfe erkennen kann. Als nächstes habe ich meine Box aus dem Vorjahr verbessert. Ein großes Problem dieser Box war, dass ich eine Kamera ohne Autofokus verwendet habe. Die Bilder waren also meist unscharf, wenn man sie nicht manuell scharf gestellt hat. Dieses Problem habe ich gelöst, indem ich die Raspberry Pi Kamera v3 verwendete (Angabe), da diese Autofokus hat. Ich verwendete die wide-version, da ich mit dieser einen größeren Bereich abdecken kann (120° statt 76°). Ich nutzte außerdem die Noir-version, damit ich die Möglichkeit habe eine Infrarot-LED zu verwenden, wenn die Bilder zu dunkel sind. Anschließend habe den Waveshare SIM7600E-H 4g-hat angebaut (Angabe), damit ich, unabhängig von freiem W-Lan, SMS-Nachrichten an einen Verantwortlichen, wie beispielsweise einen Jäger, versenden kann. Danach habe ich einen Infrarot-Scheinwerfer angebaut (Angabe), damit meine Kamera auch im Dunkeln Bilder schießen kann, andere Tiere jedoch nicht gestört werden. Da meine jetzige Box eine provisorische, nicht gut passende Box aus dem Baumarkt ist, habe ich mir eine genau passende Box 3D-drucken lassen. Ein weiteres Problem des vorherigen Systems war, dass die Erkennung sehr lange dauerte. Daher verwende ich nun den Raspberry Pi 5 mit dem Raspberry Pi AI HAT, damit ich die Bilder innerhalb weniger als einer Sekunde analysieren kann. Mit dem AI HAT habe ich außerdem die Möglichkeit ein Live-Video mit 30fps zu analysieren und in diesem alle relevanten Objekte zu umkreisen. Anschließend habe ich einige Tests mit echten Tieren gemacht. Zuerst habe ich die Kamera an meiner eigenen Katze ausprobiert (Angabe), wobei die Kamera meine Katze in fast allen Fällen als irrelevant erkannt hat. Als nächstes habe ich sie an einem Hund meines Freundes ausprobiert (Angabe). Auch bei diesem Test wurde der Hund immer erkannt. Zuletzt fuhr ich mit meiner Jugend forscht Gruppe in den Wildpark Schwarze Berge, damit ich die Kamera an echten Wölfen testen kann. Auch hier wurden die Wölfe sehr gut erkannt.

2.2 Software

2.2.1 Verwendete Module

2.2.1.1 YOLOv8

Für die Programmierung des neuronalen Netzwerkes habe ich YOLOv8 verwendet. YOLOv8 wurde von ultralytics entwickelt und steht für „you only look once“. Mit YOLOv8 kann man einfach KI-Modelle zur Segmentierung und Klassifizierung erstellen und diese mit eigenen Trainingsdaten trainieren.

2.2.1.2 Roboflow

Für das Labeln der Bilder habe ich Roboflow verwendet. Roboflow ist eine Internetseite, auf der man einfach Bilder labeln und für verschiedene Bilderkennungssoftwaren exportieren kann (Abb. 2)

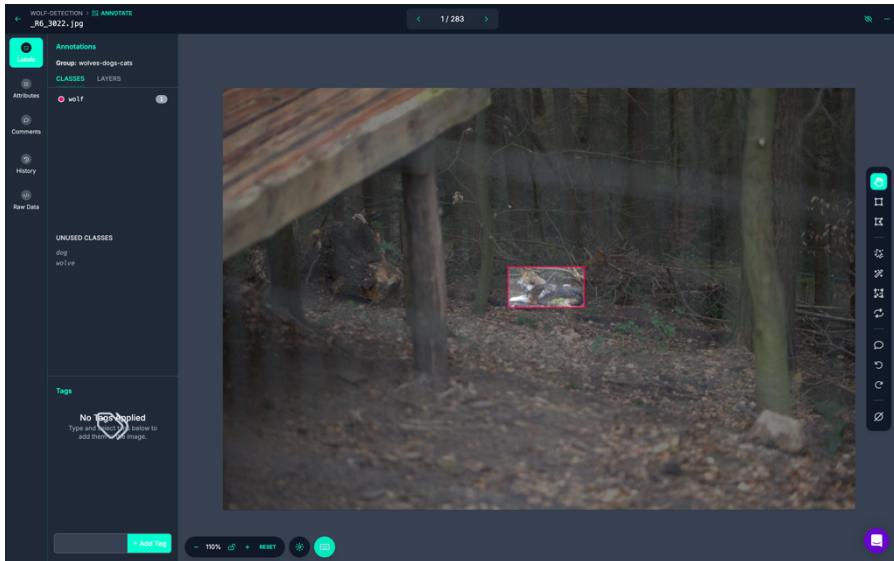


Abb. 2 Screenshot von Roboflow

2.2.2 Programmierung

2.2.2.1 Erstellung des neuronalen Netzwerkes

Ein großes Problem meiner alten KI war, dass sie die Trainingsdaten nicht gut verallgemeinern konnte. Das heißt, dass sie die Daten aus den Trainingsbildern nicht gut auf neue Bilder anwenden konnte. Ein weiteres Problem war, dass die KI nicht gut zwischen relevanten und irrelevanten Informationen unterscheiden konnte. Deshalb musste ich bei meiner alten KI erst den Hintergrund der Bilder entfernen, damit die KI nicht durch unnötige Details abgelenkt wird. Das Problem daran war, dass die Hintergrundentfernung nicht perfekt funktionierte und manchmal das relevante Tier als Hintergrund erkannt hat und ein anderes irrelevantes Objekt, welches weiter vorne stand oder größer war, als das relevante Objekt erkannte. Um diese Probleme zu lösen, habe ich mich dazu entschieden, statt TensorFlow YOLOv8 zu verwenden, da diese KI zum einen allgemein besser ist und über eine Segmentierung verfügt, das heißt, dass die KI nicht nur Bilder in verschiedene Kategorien unterteilen kann, sondern auch einkreisen kann, wo sich das Objekt genau im Bild befindet. Dies ist insofern praktisch, da die KI selbst zwischen relevanten und irrelevanten Informationen differenzieren kann und damit die Notwendigkeit für eine Vorverarbeitung nicht mehr besteht. Für das neue System musste ich jedoch auch meine Trainingsdaten anpassen, indem ich in einer separaten Datei für jedes Bild die Koordinaten des relevanten Objektes speichere.

Dies habe ich mithilfe von Roboflow (2.2.1.2) gemacht. Ich habe mich dazu entschieden die mittlere Modellgröße zu nehmen, da dieses Modell relativ viele Parameter hat, jedoch klein genug ist, sodass die Klassifizierung und das Training nicht zu lange dauern. Da mein Rechner keine starke GPU hat, würde das Training auf meinem Rechner extrem lange, wahrscheinlich mehrere Tage bzw. Wochen, dauern. Daher trainierte ich das Modell auf Google Colab, einer online Jupiter-Notebook Umgebung von Google, da das Modell dort auf deutlich leistungsstärkeren Servern von Google trainiert wird, und somit nur ca. 3 Stunden dauert. Damit das Modell auf dem läuft, musste ich es in das .hef Format konvertieren. Hef steht für „halo executable format“. Um das Modell zu konvertieren, musste ich es zuerst als .onnx Modell speichern. Anschließend musste ich es mit dem Hailo Data Flow Compiler ins .hef Format konvertieren. Dieser ist jedoch nur für Ubuntu verfügbar. Daher musste ich mir zuerst WSL installieren. WSL steht für „Windows Subsystem for Linux“. Unter WSL kann ich also auch auf Windows den Compiler installieren. Mit diesem Compiler habe ich das Modell schlussendlich konvertiert. Da das Modell während der Konvertierung automatisch optimiert wird, dauerte dies fast 13 Stunden auf meinem eigenen Rechner.

2.2.2.2 Programmierung des Codes zur Klassifizierung

Da es sehr schwierig und kompliziert ist, mit dem AI HAT zu kommunizieren, verwendete ich den offiziellen Beispielcode von Hailo und änderte diesen so ab, dass er für meinen Anwendungszweck funktioniert. Zuerst änderte ich das verwendete Modell von einem Beispielmodell zu meinem eigenen um. Anschließend änderte ich die Eingabe von einem Video um zu meiner Kamera. Ein weiteres Problem war, dass das Programm automatisch ein Fenster öffnete mit der Ausgabe der Kamera. Dies ist für Testzwecke natürlich gut, für den fertigen Gebrauch jedoch hinderlich, da es zum einen ohne ressourcenschonender ist und der Videooutput mich daran hindert das Programm über einen ssh Zugriff zu starten. Daher habe ich String, der die GStreamer Pipeline startet, so angepasst, sodass das Fenster nicht mehr geöffnet wird. Anschließend habe ich den Code so angepasst, dass die Pipeline nur gestartet wird, wenn mein Bewegungsmelder eine Bewegung erkennt und wieder gestoppt wird, sobald die Bewegung weg ist. Hierbei muss ich die Pipeline in einem anderen Thread starten, da der Code sonst unterbrochen wird. Das Schließen der Pipeline funktioniert jedoch nur im Haupt-Thread. Bevor die Erkennung startet, wird außerdem erst die Infrarot LED eingeschaltet. Des Weiteren habe ich die callback Funktion angepasst, die mit jedem Bild aufgerufen wird. Diese merkt sich nun die letzten 60 Erkennungen über 50%. Wenn auf mindestens der Hälfte dieser Bilder ein Hund, bzw. einen Wolf zu sehen ist, sendet sie mit dem SIM hat eine SMS an eine angegebene Telefonnummer.

2.3 Hardware

2.3.1 Verwendete Hardware

Für die Hardware habe ich den Raspberry Pi 5 mit dem Raspberry Pi hailo AI HAT verwendet, sowie den Raspberry Pi PIR Bewegungsmelder, ein Infrarot-LED-Array von Splenssy, ein Waveshare SIM7600E-H 4g hat und die Raspberry Pi Kamera v3 (Abb. 3)



Abb. 3 Bild der Kamerabox

2.3.1.1 Raspberry Pi Kamera v3

Bei dem letzten Versuch im Wildpark, konnte ich keine guten Bilder schießen, da meine letzte Kamera keinen Autofokus hatte. Dies ist auch gerade in einem autonomen Versuch oder im fertigen Einsatz problematisch, wenn niemand die Kamera manuell fokussieren kann. Daher habe ich mich dazu entschieden zu der Raspberry Pi Kamera v3 umzusteigen. Die

Qualität der Bilder dieser Kamera sind zwar etwas schlechter, jedoch immer gut genug und die Vorteile durch die Autonomisierung überwiegen. Die Kamera v3 hat jedoch auch Autofokus, weshalb die Bilder immer scharf sind, egal wo sich das Tier befindet.

2.3.1.2 Waveshare SIM7600E-H 4g

Ein weiteres Problem meines vorherigen Systems war, dass ich keine Möglichkeit hatte die zuständige Person zu benachrichtigen, wenn ich kein WLAN habe. Da es essenziell für mein Projekt ist, dass ich die zuständige Person alarmieren kann und im Wald, bzw. an einem Gehege höchstwahrscheinlich kein WLAN verfügbar sein wird, habe ich mich dazu entschieden den Waveshare 4g HAT zu verwenden, da ich mit diesem, egal wo der Raspberry Pi steht, SMSs versenden kann.

2.3.1.3 Infrarot-LED-Array

Damit meine Kamera auch in der Nacht sehen kann, verwende ich eine Infrarot-LED, da meine Kamera keinen Infrarot Filter hat und somit keine anderen Tiere durch das Licht gestört werden.

2.3.1.4 Raspberry Pi AI HAT

Mein vorheriges System brauchte ca. 20-30 Sekunden, um ein Bild zu verarbeiten. Da ich jedes Mal, wenn der Bewegungsmelder ausgelöst wird, mehrere Bilder schieße, läuft schnell eine große Datenmenge an Bildern an, wenn der Bewegungsmelder mehrfach hintereinander ausgelöst wird. Dies ist ein großes Problem, da so im schlimmsten Fall der Verantwortliche erst mehrere Minuten nachdem der Wolf erkannt wurde benachrichtigt wird. Der AI HAT hat 13 TOPS, dies steht für „Tera Operations Per Second“. Mit dem HAT kann ich ein Live-Video-Stream mit 30fps von meiner Kamera analysieren, d.h. ich brauche nun nur noch eine dreißigstel Sekunde zum Verarbeiten eines Bildes, was deutlich besser ist als die 20-30 Sekunden ohne AI HAT. Mit dem AI HAT habe ich nun auch die Möglichkeit ein Live-Video der Kamera zu analysieren, sobald eine Bewegung erkannt wird und nicht nur einzelne Bilder. Dies minimiert Fehlerquellen, da einzelne Bilder unscharf sein können oder dass die KI dieses Bild falsch klassifiziert. Mit einem Live-Video werden nur manche Bilder unscharf sein und es gibt immer noch genug für die KI zum Klassifizieren. Wenn die KI beispielsweise bei einem von fünf Bildern falsch liegt, würde die KI jeden fünften Wolf nicht erkennen, wenn ich bei jeder Bewegung ein Bild schieße. Wenn die KI jedoch ein Live-Video analysiert, würde bei jedem Wolf bzw. Hund der Großteil der Frames als Wolf bzw. Hund klassifiziert werden und der Alarm wird ausgelöst.

3 Ergebnisse

3.1 Versuche

Nummer	Datum	Versuch	Genaugigkeit
1	25.11.2024	Versuch mit meiner eigenen Katze	Bis auf sehr wenige Ausnahmen alles richtig
2	28.11.2024	Versuch mit eigener Katze und IR-Licht	Immer noch die meisten richtig, jedoch mehr falsch
3	04.12.2024	Versuch mit Hund	Alle erkannt
4	08.12.2024	Versuch mit Hund	Fast alle erkannt, nur wenn der Hund weit weg war, nicht
5	07.01.2025	Versuch mit Hund und Videostream	Viele erkannt, genug um einen Alarm auszulösen
6	09.01.2025	Versuch mit Wolf und Videostream	Sehr häufig erkannt
7	11.01.2025	Versuch mit Katze und fertigem System	Ein einziger Fehlalarm

Tabelle 1 Übersicht der Versuche

3.1.1 Versuche mit Katzen

3.1.1.1 Beschreibung

Um die Funktionalität meiner Kamera zu testen, habe ich sie erstmals an Katzen getestet, da ich selbst eine Katze habe und es somit relativ einfach ist das Kamerasystem an ihr auszuprobieren. Anhand dieses Testes kann ich im Nachhinein erkennen, ob meine Kamera richtig auslöst und funktionale Bilder schießt, wenn sie autonom aufgestellt ist. Um den Test durchzuführen, habe ich die Kamerabox vor dem Futterautomaten und der Katzenklappe meiner Katze aufgestellt und sie dort den Nachmittag und über Nacht stehen lassen. Diesen Test habe ich einmal mit und einmal ohne Infrarot-Licht durchgeführt.

3.1.1.2 Ergebnisse

Bei diesem Versuch hat die Kamera fast nie ein relevantes Tier erkannt (Abb. 4). Sehr selten wurde ein Hund erkannt. Dies war jedoch nur manchmal der Fall, wenn die Katze von hinten oder der Seite fotografiert wurde und wenn dies der Fall war, dann meistens mit einer Genauigkeit von ca. 30%-40% (Abb. 5), was sehr niedrig ist und nicht zu einem Alarm führen würde. Bei dem Wiederholungsversuch mit IR-Licht war die Genauigkeit etwas schlechter, die Katze wurde jedoch weiterhin in den meisten Fällen als ungefährlich erkannt. Die Genauigkeiten waren dieses Mal jedoch höher, wenn die Katze falsch erkannt wurde.

3.1.1.3 Ergebnisdiskussion

Das Kamera-System kann bereits sehr gut mit dem Bewegungsmelder scharfe Bilder schießen, die die KI für die Bild-Klassifikation nutzen kann. Die Genauigkeit des neuronalen Netzwerkes ist bereits gut, sollte jedoch für den realen noch einmal verbessert werden.



Abb. 4 Erster Test mit meiner Katze



Abb. 5 Erster Test mit meiner Katze

3.1.2 Versuche mit Hunden

3.1.2.1 Beschreibung

Um zu erkennen, ob die Kamera „gefährliche“ Tiere erkennt, habe ich dies zunächst an Hunden ausprobiert, damit auch die Differenzierung zum Wolf vorgenommen werden kann. Da ich selbst keinen Hund habe, testete ich dies an Hunden von Freunden, bzw. Bekannten. Zuerst mit dem Hund einer Freundin. Anschließend habe ich die Kamera an einem Hund von einer Nachbarin von Frau von Bargen ausprobiert.

3.1.2.2 Ergebnisse

Bei dem ersten Versuch habe ich nur wenige Bilder gemacht. Auf diesem war der Hund immer gut zu sehen und die Kamera hat den Hund immer erkannt (Abb. 6). Beim zweiten Versuch habe ich über 100 Bilder gemacht. Von diesen wurden mehr als 50% als Hund erkannt. Wenn der Hund nicht erkannt wurde, war er meistens sehr weit weg und nur klein im Bild zu sehen (Abb. 6). Wenn der Hund näher dran war, wurde er auch fast immer erkannt (Abb. 7).

3.1.2.3 Ergebnisdiskussion

Die Genauigkeit, wenn der Hund relativ nah an der Kamera ist, ist bereits sehr zufriedenstellend. Wenn der Hund jedoch weit weg ist, könnte die Genauigkeit besser sein. Dies liegt höchstwahrscheinlich an den Trainingsdaten. Da diese Bilder von Hunden aus dem Internet sind, habe ich nur sehr wenige Bilder von Hunden im Hintergrund unter den Trainingsdaten. Um die Genauigkeit weiter zu verbessern, werde ich mehr Bilder von Hunden, die weiter entfernt sind in den Trainingsdaten verwenden.

3.1.3 Versuch mit Hund und Videostream

3.1.3.1 Beschreibung

Nachdem ich den AI HAT hatte und das neue Programm mit einem Videostream programmiert hatte, habe ich die KI an einem weiteren Hund getestet. Dieses Mal wurden alle Erkennungen direkt in den Kamera-Stream gezeichnet, was es deutlich einfacher machte. Ich habe für diesen Versuch nicht mein Hauptprogramm verwendet, da für Testzwecke ein einfaches Programm, welches einfach einen Videostream mit allen Erkennungen öffnet, deutlich praktischer ist.



Abb. 6 Erster Test mit einem Hund



Abb. 7 Erster Test mit einem Hund

3.1.3.2 Ergebnis

Die KI hat den Hund nicht immer erkannt (Abb. 8), jedoch sehr oft (Abb. 9). Dies war zum einen davon abhängig, wie gut der Hund zusehen war, falls er beispielsweise von Möbeln verdeckt wird oder in welche Richtung er sich bewegt hat, d.h. wenn er sich beispielsweise von der Kamera weg bewegte, wurde dies oft nicht erkannt, wenn er auf sie zu kam jedoch deutlich häufiger. Zum anderen gab es jedoch natürlich auch einzelne Bilder, die nicht erkannt wurden, weil dieses Bild unscharf war.



Abb. 8 Test mit Hund und Videostream

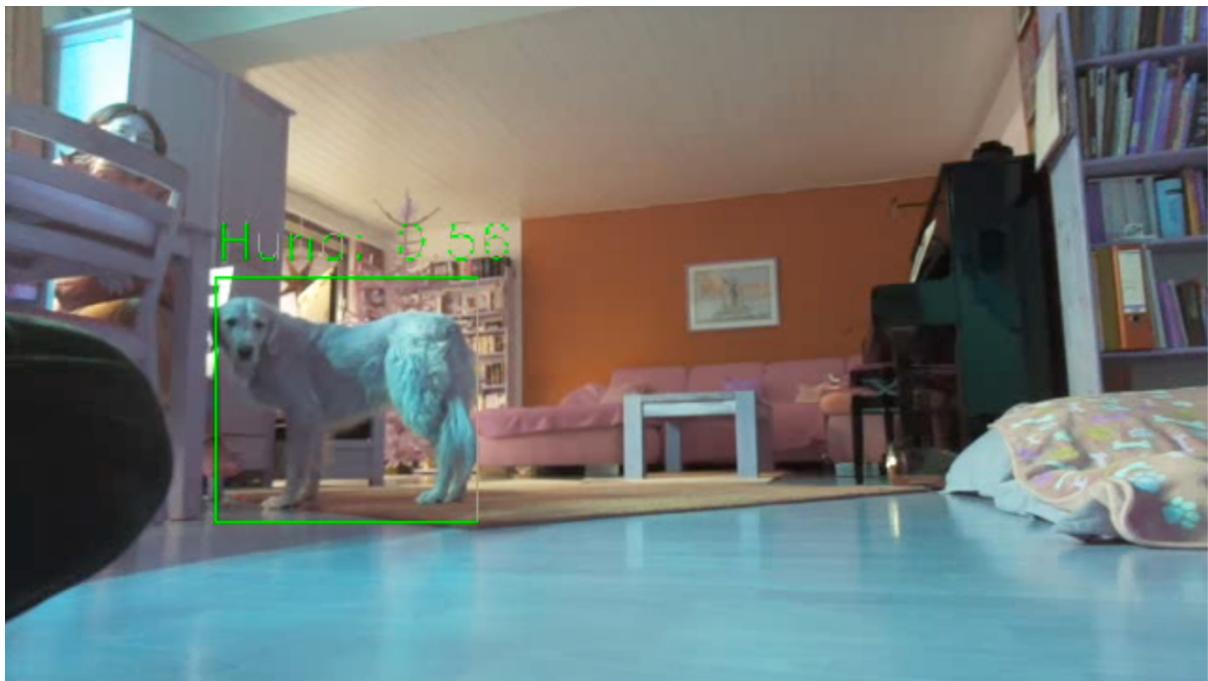


Abb. 9 Test mit Hund und Videostream

3.1.3.3 Ergebnisdiskussion

Die Genauigkeit meiner KI in diesem Versuch ist sehr zufriedenstellen. Bei der sehr großen Gesamtmenge an Bildern in dem Versuch war es nicht zu erwarten, dass die KI den Hund jedes Mal erkennt, und es wäre definitiv ein Alarm ausgelöst worden.

3.1.4 Versuche mit Wölfen und Videostream

3.1.3.1 Beschreibung

Um auszuprobieren, ob meine KI auch mit echten Wölfen funktioniert und nicht nur mit Hunden, bin ich in den Wildpark Schwarze Berge gefahren, um dort meine Kamera an den Wölfen auszuprobieren. Vor Ort hat mir die Wolfsberaterin Svenja

Oßenbrügge geholfen, indem sie die Wölfe angelockt hat, da das Gehege sehr groß ist und ich ansonsten keine verwendbaren Bilder der Wölfe bekommen würde.

3.1.3.2 Ergebnis

Da meine Kamera bei dem Wildparkbesuch letztes Jahr nicht gut funktioniert hat, war ich gespannt, ob meine Optimierungen zu Erfolg führen. Tatsächlich hat die KI den Wolf fast immer erkannt. Interessant war, dass der Wolf, wenn er durch das Bild gelaufen ist, immer erkannt wurde (Abb. 10), wenn er jedoch ruhig in der Mitte lag, bzw. teilweise auch stand, wurde er weniger häufig erkannt (Abb. 11).



Abb. 10 Test mit Wolf und Videostream



Abb. 11 Test mit Wolf und Videostream

3.1.3.3 Ergebnisdiskussion

Die Genauigkeit von meiner Kamera, mit echten Wölfen ist mehr als zufriedenstellend. Das der Wolf nicht erkannt wird, wenn er auf dem Boden liegt, liegt daran, dass solche Bilder nicht, bzw. nur sehr selten in meinem Trainingsdaten gab. Dass er dann nicht erkannt wird, ist jedoch nicht schlimm, da es zum einen sehr unwahrscheinlich ist, dass der Wolf, sich vors Schafgehege legt. Sollte er dies doch tun, muss er jedoch auf jeden Fall vorher erst einmal dort hingehen, wobei der dann von der Kamera erkannt wird. Bei einem Wolfsriss sind die Wölfe aktiv und bewegen sich schnell, daher ist die Erfolgsquote sehr gut. Diese werden von meinem Kamerasytem durch den Bewegungsmelder aktiviert und aufgenommen, damit dann durch die KI der Wolf als solcher erkannt wird. Ich stelle fest, dass das System unter Bedingungen, die so realitätsnah wie möglich simuliert wurden, ihr Ziel erreicht – das Erkennen von Wölfen in naturnahen Geländen.

3.1.4 Versuche mit Katze und fertigem System

3.1.3.1 Beschreibung

Um mein neues System zu testen, stellte ich meine Kamerabox ein weiteres Mal vor der Katzenklappe und Futterbox meiner Katze auf. Mit diesem Versuch möchte ich zum einen herausfinden, ob mein System richtig funktioniert, das heißt, ob der Bewegungsmelder richtig ausgelöst wird, der Videostream gestartet wird, das IR-Licht angeht, und so weiter. Dieser Versuch dient außerdem als Kontrollversuch, das heißt, um zu sehen, ob meine Kamera bei einem für das Projekt irrelevantem Tier nicht auslöst.

3.1.3.2 Ergebnis

Meine Kamera hat immer ausgelöst, wenn die Katze vorbeigelaufen ist. Einige Male wurde das System auch ausgelöst, wenn sie nicht vorbeigelaufen ist, das lag dann jedoch daran, dass der Bewegungsmelder eine von uns verursachte Bewegung wahrgenommen hat. Die KI hat meine Katze ein paarmal als Hund eingekreist (Abb. 12 u. Abb. 13), da mein Programm jedoch nur eine SMS sendet, wenn viele Bilder hintereinander als Hund klassifiziert wurden, habe ich nur eine Warn-SMS bekommen.

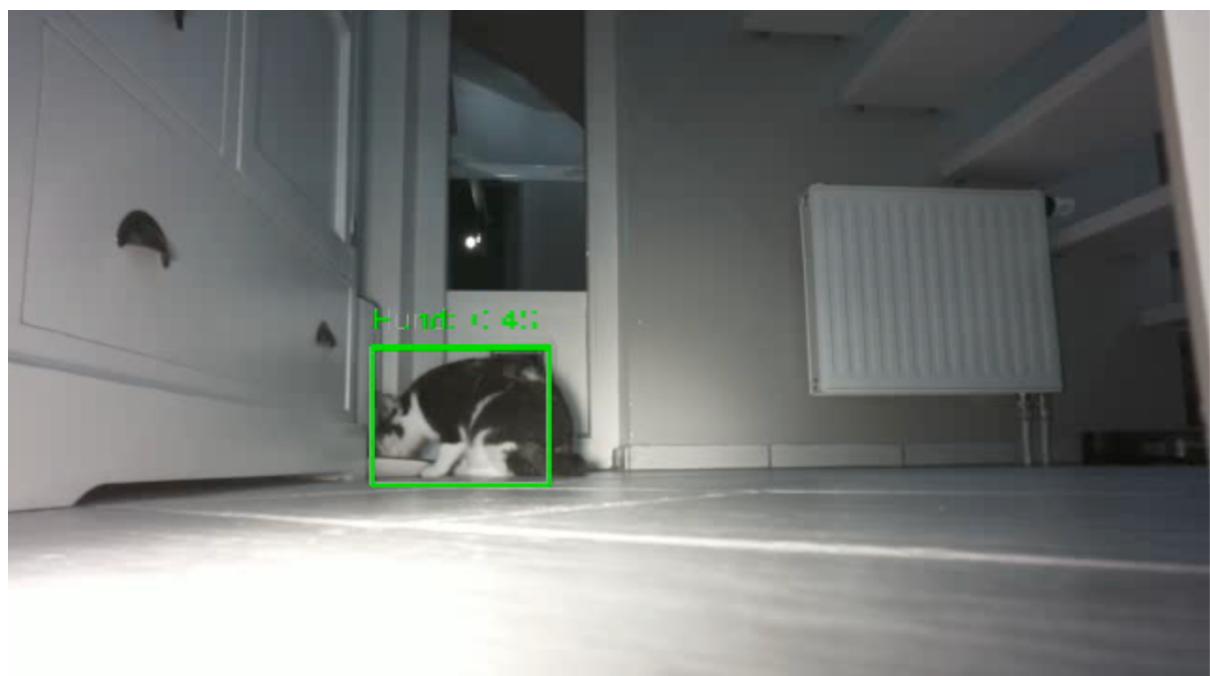


Abb. 12 Test mit Katze und Videostream



Abb. 13 Test mit Katze und Videostream

3.1.3.3 Ergebnisdiskussion

Mein System hat, abgesehen von der Erkennung, perfekt funktioniert. Die Erkennung war auch schon sehr gut, das einzige Schwierige für die KI war, die Katze als unwichtig zu erkennen, wenn sie von der Kamera weglief. Dies liegt daran, dass in meinen Trainingsbildern nur wenige Katzenbilder aus dieser Perspektive sind. Um das System noch weiter zu verbessern, werde ich also weitere Bilder von Katzen von hinten den Trainingsbildern hinzufügen.

4 Ergebnisse und Zielerreichung

4.1 Zielerreichung

Mein Ziel war es, zuerst das neuronale Netzwerk aus dem vorherigen Jahr mit YOLOv8 zu verbessern. Dies ist mir gelungen. Mein neues Netzwerk verfügt nun über eine optimierte Segmentierung. Dieses zeigt sich insbesondere an den Testdurchläufen im Wildpark Schwarze Berge. Das Gelände ist abschüssig und in Hanglage, umgeben von hohen Bäumen, die Dunkelheit und Lichtwechsel verursachen. Der Test fand am 09.01.2025 gegen 14:30 Uhr statt, wenn die Sonne bereits wieder tiefer steht, da der Sonnenuntergang ab 16:30 Uhr ist. Auch unter diesen widrigen Umständen, die jedoch in der freien Wildbahn vorkommen können, hat mein Kamerasystem die Wölfe aufgenommen und die KI diese erkannt. Der Alarm wurde zu diesem Zeitpunkt ausgestellt, da ich während des Tests keine Wölfe irritieren wollte, jedoch läuft dieser nach dem Wiedereinschalten einwandfrei. Durch diese Testdurchläufe konnte ich so realitätsnahe Testbedingungen simulieren, wie es für mich möglich war. Ein Test in einer freien Wildbahn ist aufgrund der geringen Wahrscheinlichkeit eines Tierrisses in dem von mir ausgewählten Gelände schwer möglich.

Ich habe außerdem eine individuelle Box für meine Hardware. Diese konnte ich auch verbessern, da ich im Vorjahr nur in der Lage war, alle 20 – 30 Sekunden ein einzelnes Bild zu analysieren und nun, mit meiner neuen Hardware ein 30fps Video-Stream live analysieren kann.

Ich kann nun außerdem, unabhängig vom WLAN, eine SMS an den Verantwortlichen senden, da ich meine Hardware um ein Simkarten-Modul ergänzt habe.

4.2 Ergebnisdiskussion

Das Ziel meines KI- und Kamerasystems ist, Tierhaltern wie Schafhirten zu helfen, indem diese rechtzeitig gewarnt werden können, wenn ein Hund oder Wolf sich ihrer Schafherde nährt. So haben diese die Chance den Angreifer zu vertreiben, beispielsweise mit einem Warnschuss, bevor er die Tiere reißen kann und somit ihre Existenz zu sichern und ihre Tiere zu schützen, die sie lieben. Zum anderen ist es jedoch auch sehr wichtig für die Wolfberater und auch für die Betroffenen bei einem Wild Riss zu wissen, ob dieser von einem Hund oder Wolf verursacht wurde und genaue Bilder des Tieres zu haben. Es ist auch sehr wichtig mehr über die Wölfe und deren Verhaltensweisen zu wissen, um die Wölfe und die Nutztiere des Menschen zu schützen. Im Moment liegen bei fast allen Wolfsrissen keine Bilder, bzw. kein Video des Tieres vor und die Wolfsbeauftragten müssen ausschließlich anhand des toten Tieres feststellen, ob es sich um einen Hund oder Wolf handelte. Diese stellen fest, dass Nutztiere zu ungefähr zu 50% von Wölfen und zu 50% von Hunden gerissen werden. Die Hinweise ergeben sich hierbei im Moment nur aus DNA-Proben, vor allem aber der Vorgehensweise des Angreifers. Daher ist die Wolfberaterin Svenja Oßenbrügge auch der Meinung, dass Videoaufnahmen wichtig sind, um eindeutig festzustellen, ob es sich um einen Hund oder Wolf handelte. Das System würde außerdem mögliche Fehlerquellen durch Fehleinschätzungen minimieren. Eindeutige Beweise würden außerdem das Wolfsmonitoring verbessern.

Ich alle Komponenten des Systems einzeln testen. Das autonome Auslösen des Systems durch den Bewegungsmelder konnte ich mithilfe meiner Katze ausprobieren, indem ich das Kamerasystem vor der Futterbox aufstellte und die Kamera immer, wenn die Katze essen ging, auslöste. Da meine KI sowohl Hunde als auch Wölfe erkennt, konnte ich sehr erfolgreich mit echten Tieren bestätigen. Da meine Kamera ein Video aufnimmt und analysiert, sobald eine Bewegung erkannt wird und die KI echte Hunde und Wölfe erkennen kann, würde meine Box sehr wahrscheinlich auch im echten Gebrauch funktionieren.

4.3 Ausblick

Mein Projekt ist jetzt so gut wie gebrauchsfertig. Das Einzige, was ich noch verbessern kann, sind weitere Bilder den Trainingsbildern hinzuzufügen, um die Genauigkeit der KI weiter zu steigern. Das heißt, ich werde noch weitere Bilder von Katzen und anderen nicht relevanten Tieren aus unterschiedlichen Blickwinkeln und mehr Bilder von Hunden und Wölfen den Trainingsbildern hinzufügen. Abgesehen davon ist mein System fertig und könnte so für den Schutz der Nutztiere des Menschen verwendet werden.

5. Quellen und Funktionen der Autoren

5.1 Quellen

Stader Tageblatt, Grit Klempow, Lauter Ruf nach Wolfsabschüssen, Datum: 25.01.2023

Harishvutukuri, Dogs vs wolves, URL: <https://www.kaggle.com/datasets/harishvutukuri/dogs-vs-wolves>, Datum: 2020, Datum des Aufrufs: 04.01.2024

Imagenet, URL: https://images_cv/dataset/white-wolf-image-classification-dataset, Datum des Aufrufs: 04.01.2024

Shaunthesheep, Microsoft Cats vs Dogs, URL: <https://www.kaggle.com/datasets/shaunthesheep/microsoft-catsvsdogs-dataset>, Datum: 2020, Datum des Aufrufs: 04.01.2024

Zörb, Paul, mein GitHub-Projekt, URL: <https://github.com/Paul193247/Jugendforsch-Wolfserkennung>

5.2 Abbildungen

Abb. 1 Titelbild, Institut für Ökologie und Evolution, Universität Bern, URL: <https://infoplattform-grossraubtiere.ch/bei-ihrer-rueckkehr-achten-woelfe-auf-das-beuteangebot/>, heruntergeladen am 10.01.2025

Abb. 2 Screenshot von Roboflow, eigenes Bild (Paul Zörb)

Abb. 3 Bild der Kamerabox, eigene Aufnahme (Paul Zörb)

Abb. 4 Erster Test mit meiner Katze, eigene Aufnahme (Kamerabox)

Abb. 5 Erster Test mit meiner Katze, eigene Aufnahme (Kamerabox)

Abb. 6 Erster Test mit einem Hund, eigene Aufnahme (Kamerabox)

Abb. 7 Erster Test mit einem Hund, eigene Aufnahme (Kamerabox)

Abb. 8 Test mit Hund und Videostream, eigene Aufnahme (Kamerabox)

Abb. 9 Test mit Hund und Videostream, eigene Aufnahme (Kamerabox)

Abb. 10 Test mit Wolf und Videostream, eigene Aufnahme (Kamerabox)

Abb. 11 Test mit Wolf und Videostream, eigene Aufnahme (Kamerabox)

Abb. 12 Test mit Katze und Videostream, eigene Aufnahme (Kamerabox)

Abb. 13 Test mit Katze und Videostream, eigene Aufnahme (Kamerabox)

5.3 Tabellen

Tabelle 1 Übersicht der Versuche, eigene Darstellung (Paul Zörb)

6. Danksagung

An dieser Stelle möchte ich mich bei all denjenigen bedanken, die mich bei meinem Projekt unterstützt und motiviert haben. Zuerst gebührt mein Dank Herrn Privatdozent Dr. Carmesin, der mein Projekt betreut und mir bei der Organisation des Projektes geholfen hat. Ein besonderer Dank gilt Frau von Bargen, die mir mit viel Geduld, Interesse und Hilfsbereitschaft insbesondere beim Schreiben der Langfassung zur Seite stand. Zu guter Letzt möchte ich mich bei Florian von Bargen und meiner Familie bedanken, die stets ein offenes Ohr für mich hatten. Ich möchte mich außerdem bei Svenja Oßenbrügge, Wolfsberaterin Landkreis Hamburg-Harburg, und dem Wildpark Schwarze Berge für die vertiefenden, weiterführenden Informationen zum Riss von Weide- und Nutzieren und für die Unterstützung bei der Wolfsbeobachtung am Wolfsgehege des Wildparks Schwarze Berge bedanken.