

37.OTDK Biológia Szekció

A paradicsomhalak jelleme: mélytanulási algoritmusok alkalmazása komplex
viselkedésformák non-invazív detektálására

Csáki Dominika, Alternatív Közgazdasági Gimnázium 13.évf.

Kutató Diákok Országos Szövetsége (KUDOSZ)

Témavezető: Dr. Varga Máté, ELTE Genetika tanszék

Tartalom

1. BEVEZETŐ	2
2. CÉLKITŰZÉS	3
3. ANYAGOK ÉS MÓDSZEREK	4
3.1 Felvételek elkészítése és a használt kamerák paraméterei	4
3.2 Kvantifikáció.....	4
3.2.1. Reprezentatív képkockák kiválasztása kézi címkézéshez	5
3.2.2. Testpontok definiálása és felcímkézése (labeling).....	5
3.2.3. A későbbi értékelési fázisban használatos („teszt”) részadathalmaz leválasztása ...	5
3.2.4. Tréning fázis	6
3.2.5. Értékelési fázis (evaluáció).....	6
3.2.6. A modell pontosítása.....	7
3.2.7. Videók elemzése a betanított modell alkalmazásával.....	7
3.3 Klasszifikáció	8
3.4. Kvantifikáció és klasszifikáció kombinálása	9
3.5. Háromdimenziós kvantifikáció	9
3.5.1 Kalibráció	10
3.5.2 Háromszögelés (2D ->3D)	11
4. EREDMÉNYEK.....	12
4.2. Felszíni levegővétel részletes elemzése.....	12
4.2.1. Hát görbületének vizualizációja.....	12
4.2.2. Egyéb paraméterek vizsgálata	14
4.2.3. Korrekciós megfontolások kézi elemzéssel	15
4.2.4. Konklúzió	16
5. DISZKUSSZIÓ	17
5.1 Kvantifikációs megfontolások	17
5.1.1 Labeling konzisztenciájának javítása	17
5.1.2. Háromdimenziós kvantifikáció szükségessége	20
5.2. Klasszifikáció	20
5.2.1. A klasszifikáció szükségessége.....	20
5.2.2. A választott klasszifikációs stratégiák indoklása	21
Köszönetnyilvánítás	23

1. BEVEZETŐ

A kínai paradicsomhalat (*Macropodus opercularis*) széles viselkedési-repertoár jellemzi, melynek jellegzetes elemei a bonyolult párzási koreográfia, a territoriális viselkedés és a felszíni levegővétel. Utóbbi viselkedésforma igazi különlegessége a labirintkopoltyús halakat is magába foglaló *Anabantoidei* alrendnek: az ide sorolható halaknál a hagyományos kopoltyúval történő oxigénfelvétel az ún. labirintusszerv működésével egészül ki, lehetővé téve, hogy a halak a légköri oxigént is hasznosítsák. Ennek köszönhetően ezek a fajok képesek a túlélésre akár hipoxiás (alacsony oxigéntartalmú) környezetben is. Felszíni levegővétel során a hal a vízfelszínre úszik levegőért, amely a szájüregén keresztül a kopoltyúk fölött kialakult, ún. szuprabranchiális üregbe (SBC) jut, ahol az üreget borító, kapilláris-dús epitélium, illetve az itt elhelyezkedő, hasonló epitéliummal borított a labirintusszerv segítségével történik meg az oxigén felvétele a bejutott levegőből. [1]

Etológiai kutatásokban a kínai paradicsomhalat már régóta használják modellszervezetként, és bár az előző évszázad utolsó évtizedeiben az Eötvös Loránd Tudományegyetem (ELTE) Etológia Tanszékén dolgozó Csányi Vilmos professzor csoportja végzett részletes viselkedési megfigyeléseket [2], az akkor elérhető eszköztár miatt számos kérdés megválaszolására egyszerűen nem volt lehetőség. Ennek megfelelően azt is csak nagyon korlátozottan lehetett vizsgálni, hogy a halak kódolt magatartásformáinak milyen genetikai gyökerei vannak. [3]

A normális viselkedés jellemzéséhez, illetve az attól való eltérések pontos elemzéséhez alapvetően kétféle adat begyűjtésére van szükségünk: egyfelől pozicionális adatokra (kvantifikáció), ami az állatok mozgásának lekövetésével valósulhat meg. Sokáig erre legfeljebb az élőlények testéhez erősített érzékelőkkel volt lehetőség [4], azonban ez a beavatkozás egy invazív módszer, így maga is a viselkedés változását eredményezheti. Másrészt, a pontos viselkedésforma meghatározása a vizsgált viselkedési periódus bármely pillanatában (klasszifikáció). Ez a „kódolás” korábban manuálisan történt, de sok adat feldolgozása nagyon időigényes lehet és pontosságát meglehetősen befolyásolhatja az emberi tényező [5].

A mesterséges intelligencia (MI) térnyerése gyorsabb és részletesebb elemzést nyújtó alternatívát kínál mindkét fenti módszerre, és a megfelelő mélytanulási (ML) algoritmusok

alkalmazásával a munkafolyamatok szinte teljesen automatizálhatók válnak. Ezt a technológiát már sikeresen alkalmazták több állatfaj viselkedésének vizsgálatában is [6, 7, 8], ezzel megerősítve a módszer hatékonyságát.

A paradicsomhalak viselkedés-kutatása esetében az ML-alapú megközelítések kifejlesztése különösen indokolt lenne, hiszen a faj komplex viselkedései és jól dokumentált etogramja kiváló alapot biztosít az MI-n alapuló elemzésekhez. Ez nemcsak a viselkedési mintázatok pontosabb felismerését és kvantifikálását tenné lehetővé, hanem lehetőséget kínálna a fent említett genetikai implikációk részletesebb vizsgálatára is. Mindez hozzájárulna a viselkedésben bekövetkező változások finomabb, eddig rejtve maradt részleteinek feltárásához.

2. CÉLKITŰZÉS

A kutatás során az ELTE Genetika Tanszék Halgenetika Kutatócsoportjával [9] dolgozom együttműködésben, ahol munkám hosszútávú célja egy olyan rendszer kiépítése (azaz mind a szükséges hardware-, mind a szükséges szoftverkörnyezet kialakítása), amivel a paradicsomhalak teljes viselkedési-repertoárjának automatikus videóalapú elemzése valósulhat meg. A cél eléréséhez túlnyomórészt nyílt forráskódú ML-algoritmusokat használtam fel (pl. DeepLabCut [10], LabGym [11]), illetve ezek kimeneti adatsorait kombináltam.

Az első kisebb, de konkrét cél egy viszonylag egyszerű magatartásforma, a felszíni levegővétel részletes leírása, pontosabban a hát görbületének vizsgálata a viselkedés során.

Előzetes megfigyelés alapján a paradicsomhal felszíni levegővétel során „homorít”, azonban ennek alátámasztásához megfoghatóbb adatokra van szükség. Egyetlen halat követő videókon 27 (ebből 17 háti) testpont kerül lekövetésre (DeepLabCut algoritmus), a felszíni levegővétel detektálásáért pedig a viselkedés jellegzetes sziluettje miatt egy nyers képi adatokkal dolgozó, felügyelt (supervised) tanulás útján formált klasszifikációs modellt alkalmaztam (LabGym). Az így begyűjtött, kombinált adatokkal pontosabban lehet jellemezni a testtartást a levegővétel során, és ezek az adatok lehetővé teszik a részletes vizualizációt is.

3. ANYAGOK ÉS MÓDSZEREK

3.1 Felvételek elkészítése és a használt kamerák paraméterei

A vizsgálat első lépése minden esetben a videófelvevételek elkészítése volt a halról (2A ábra) laboratóriumi környezetben, egy 45.9 x 45.9 x 22 cm-es akváriumban. A felvételeknél fontos a neutrális háttér (pl. sötét anyag használatával) és a megfelelő világítás, hogy a hal jól elkülönüljön a környezetétől.

Az elemzésekhez használt kamerák: egy Canon 700D (62 mm) és egy Canon EOS M50 (67 mm). Mindkét kamera 1920 x 1080 pixeles felbontásban, 50 FPS sebességgel működött.

Bár ezek az eszközök nem kifejezetten kutatási célokra készültek, és nem rendelkeznek erre a legideálisabb paraméterekkel, az elemzések sikeresen lebonyolításra kerültek, és nem tapasztaltunk ebből eredő hátrányokat. Számomra az volt a leglényegesebb, hogy megfelelő számú képkockát rögzítsek ahhoz, hogy a hirtelen pozícióváltozások is pontosan követhetők legyenek. Ugyanakkor figyelembe kell venni, hogy több képkocka nagyobb adathalmazt eredményez, ami az elemzések feldolgozási idejét növelheti: a szakirodalom szerint a 100 FPS feletti képfrekvencia ebből kifolyólag teljesen szükségtelen [12]. Fontos az is, hogy a rögzített anyag adott időegységre nézve elérje azt a kritikus képkockaszámot, amely minimalizálja az utólagos, nem hardveres szinkronizáció miatt esetlegesen felmerülő „képkockák közötti” elcsúszásokat. Emellett a sűrűbb adatállomány a már említett Gaussian simítás (smoothing) hatékonyságát is javítja, ami tovább növeli az elemzések megbízhatóságát.

3.2 Kvantifikáció

Bár már többféle, nyílt forráskódú algoritmus is elérhető (mint pl. SLEAP [13]), azonban a legelterjedtebb a DeepLabCut (DLC). Azért is esett erre az algoritmusra a választásunk, már a munkám elején, mert számos korábbi publikációban, több különböző faj esetében sikeresen alkalmazták, ami arra engedett következtetni, hogy könnyen adaptálható lesz a mi céljainkra is. A DLC továbbá aktív fejlesztői közösséggel is rendelkezik, amelynek tagjai folyamatosan fejleszti és javítja a programcsomagot. Önmagában a széles felhasználói köre miatt is egyre stabilabb ez a program, a rendszerspecifikus hibái pedig egyre ritkábbak.

A kvantifikálásra alkalmas DLC modell betanításának lépései a következők voltak:

3.2.1. Reprezentatív képkockák kiválasztása kézi címkézéshez

Egy változatos, a teljes adathalmazról reprezentatív képkockákból álló adatállományt kell kiválasztani az elkészült videókból (2B ábra). Ez a lépés kulcsfontosságú, hiszen az MI csak azt fogja tudni reprodukálni, amit megtanult, és ideális esetben ebből tud általánosítani.

3.2.2. Testpontok definiálása és felcímkézése (labeling)

A lekövetendő testpontokat a konzisztencia megőrzése érdekében pontosan definiálni kell (2C ábra), majd a kiválasztott képkockákon ezeket a testpontokat fel kell címkézni (2D ábra). A munkámhoz használt felvételek esetében összesen (több iteráción át) 754 képkocka került manuálisan felcímkézésre.



2. Ábra, A tréning adathalmaz létrehozásának sematikus illusztrációja: miután a halról elkészültek a felvételek (A), változatos képkockák kerültek kiválasztásra (B), definiálásra kerültek a lekövetendő testpontok (C) majd felcímkézésre az imént kiválasztott képkockák mindegyikén (D).

3.2.3. A későbbi értékelési fázisban használatos („teszt”) részadathalmaz leválasztása

Az előző elvek szerint felcímkézett képkockák 5%-a leválasztásra került, és nem képezte részét a tanító adathalmaznak (training dataset), így nem vett részt a modell betanításában sem. Ezek ugyanakkor fontos frakcióját képezik a későbbi evaluációs fázisban használt adatoknak, mivel olyan képkockákat tartalmaz, amelyeken a testpontok pontos helyzetét a felhasználó manuálisan megjelölte. Ugyanakkor ezek az adatok a modell szempontjából újként kezelhetők, hiszen soha nem találkozott velük a tanulási folyamat során. Ezáltal lehetőségünk nyílik

kiszámítani a modell által becsült testpontok és azok manuálisan meghatározott „valós” helyzete közötti, pixelben mért távolságot, így a modell pontossága (pixelben mérve) kiértékelhető.

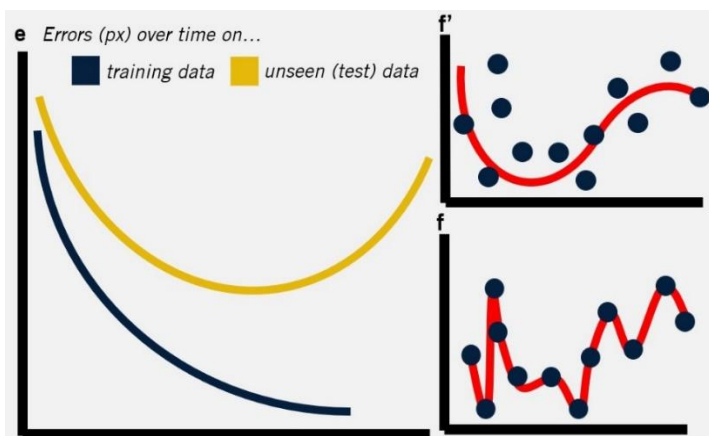
3.2.4. Tréning fázis

A tréning fázis több iteráción át, Google Colab felületén (<https://colab.research.google.com>), Graphics Processing Unit (GPU) gyorsítással zajlott, és hatékonyságának maximalizása érdekében számos kimeneti adatot (output) figyelembe kellett venni. Ilyen például a tanulási sebesség (learning rate), amely meghatározza, hogy a modell milyen gyorsan igazítja a paramétereit: túl nagy érték instabilitást, túl kicsi érték lassú konvergenciát eredményezhet. Ehhez kapcsolódik veszteségfüggvény (loss) érték változásának figyelése is, amely azt mutatja meg, hogy a modell tényleg tanul-e: itt kiszűrhető az ún. tanulási fennsík (learning plateau, ellaposodás) is, amely azt jelzi, hogy a modell teljesítménye már egyáltalán nem javul szignifikánsan, és további training csupán overfittings modellhez fog vezetni. [14]

Az általam alkalmazott modell összesen 480 ezer iteráción keresztül tanult (9-szer ~50 ezer iterációnként haladtunk, ezt követte mindig az evaluációs fázis, illetve az új, címkézett képkockák hozzáadása).

3.2.5. Értékelési fázis (evaluáció)

Minden tanulási ciklust egy evaluációs fázis követ. Ennek során kiszűrhető például a „túlillesztés” (overfitting) jelensége is (3.ábra): ilyenkor a pixelben mért hiba (error) a tréninghez felhasznált adathalmazon sokkal alacsonyabb lesz, mint az ismeretlen adatokon [14]. A másik véglete a problémának pedig az „alulillesztés” (underfitting), amikor nem ismer fel finomabb jellemzőket, mert még túl általános a modell [15]. Mindkettőt fontos kiküszöbölni egy jól általánosító, de egyben megfelelő specificitást mutató modell érdekében.



3. Ábra: Az overfitting jelensége. Egy jól általánosító (f) és egy túlillesztett (f') modell paramétereinek alakulása. A pixelben mért hibák változása a teszt- és tréning adatokon a tanulás előrehaladtával (e): az overfittinges modell a tanulási fázisban látott (tréning) adatokon sokkal kevésbé hibázik, mint az ismeretlen (teszt) adatokon. Ez az evaluációs fázis során felismerhető: egy jól általánosító modell (g) és egy overfittinges modell (g') evaluációs összesítése: bár a jól általánosító modell is jobban teljesít a látott adatokon, mint a teszt adatokon, de túlillesztettnél feltűnőbb a különbség.

3.2.6. A modell pontosítása

Ha a modell pontossága nem érte el a kívánt értéket, többféle lépés tehető a probléma megoldásáért. Az egyik az újratranítás (retraining), amely során a modell paramétereit érdemes finomítani és a tréninget korábban leállítani (early stopping), ezzel minimalizálva az overfitting kockázatát. Kulcsfontosságú ellenőrizni a felcímkézett adatok minőségét is, hogy azonosíthassuk a címkézés során esetlegesen előforduló hibákat (a begyűjtött adatok vizualizálhatóak is az eredeti képkockákon).

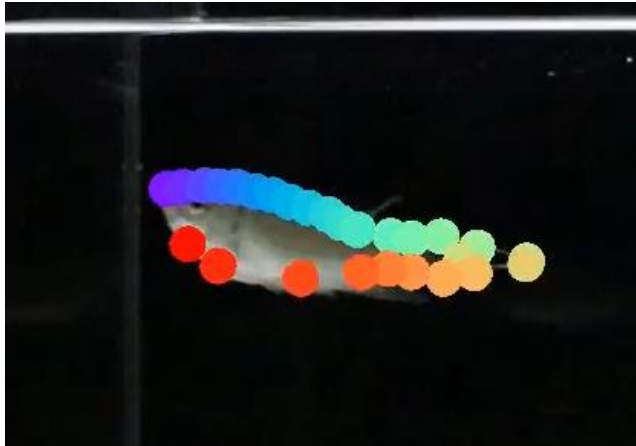
Gyakran előfordul, hogy a modell teljesítményének javításához további adatokra van szükség. Ezek lehetnek új, korábban nem használt adatok (pl. új videók elemzéséből származók), vagy az adathalmazt mesterségesen növelhetjük adat augmentációval. Ez utóbbi során például forgatás, tükrözés vagy más transzformációk, változtatások révén hozunk létre új adatokat az „eredetiek” alapján, ami robosztusabbá teszi a modellt. [16]

Esetenként utólagos optimalizációs technikák is felhasználhatóak. Ilyen például a Gaussian-simítás (smoothing) segítségével csökkenthető a zaj miatt keletkezett inkonzisztenciák az adatokban azáltal, hogy a szomszédos adatpontok átlagolódnak, így kisimítva az „ugráló” értékeket. [17]

3.2.7. Videók elemzése a betanított modell alkalmazásával

Az evaluációs fázis alapján jól teljesítő, betanított modell így alkalmassá válik még nem látott, de hasonló környezetű paradicsomhalas videók elemzésére is. A végeredmény, hogy hasonló

adathalmazok esetében azoknak csak kis százalékát kell manuális munka árán felcímkézni, mert az adatok túlnyomó többségében a modell a címkézési feladatot már át tudja venni (4.ábra).



4. Ábra: Egy tipikus kvantifikációs elemzés vizualizációjából kimetszett képkocka. A pontok a fájlban tárolt képkockákhoz (azon belül testpontokhoz) tartozó x és y koordinátákat ábrázolják, lehetőséget adva a modell pontosságának újbóli ellenőrzésére. Ez alapján következtetéseket vonhatunk le arról, hogy milyen típusú példákon hajlamos hibázni, és ennek megfelelően bővíthetjük a tréning adathalmazt.

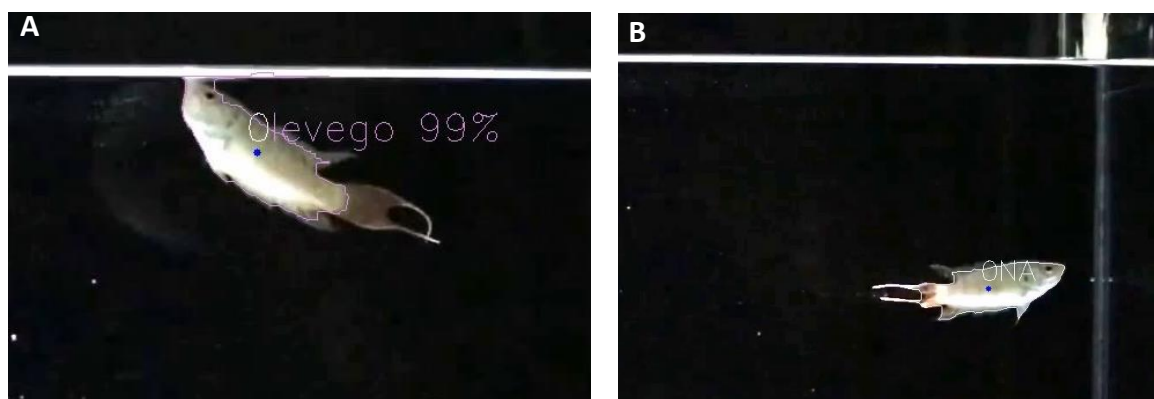
Munkám során összesen 10 felvételt (5 párt) elemeztem, amelyek szinkronizáltan (50 FPS és 1080x1920 pixeles felbontással) működtek, lehetőséget teremtve a későbbi 3D elemzésekhez (lásd később). Emellett 4 további felvételt is vizsgáltunk (átlagosan egyenként ~5 percesek, 25 FPS és 1080x1920-as felbontásúakat).

Így bruttó több mint 100 000 képkockát elemeztem, amelyből mindössze 754-et használtunk a betanításhoz, ami a teljes képkockaszám 1%> -át teszi ki. A felvétel-párok kombinációja révén az elemzések körülbelül 30 percnyi rögzített adatot fedtek le. A modell az utolsó iteráció (9., összesen 480 000 it) evaluációs fázisában új adatokon (test data) 3.29 pixel pontossággal teljesített.

3.3 Klasszifikáció

A felszíni levegővétel detektálásához felügyelt és képi-alapú LabGym modell került alkalmazásra. A modell betanításának lépései csak a jelölési fázisban térnek el a kvantifikáló modellétől: itt „példákat” (szakaszokat) kell felcímkézni aszerint, hogy látható rajta felszíni levegővétel (1) vagy nem (0).

A betanított klasszifikációs modellel analizáltuk a videókat (5. ábra).



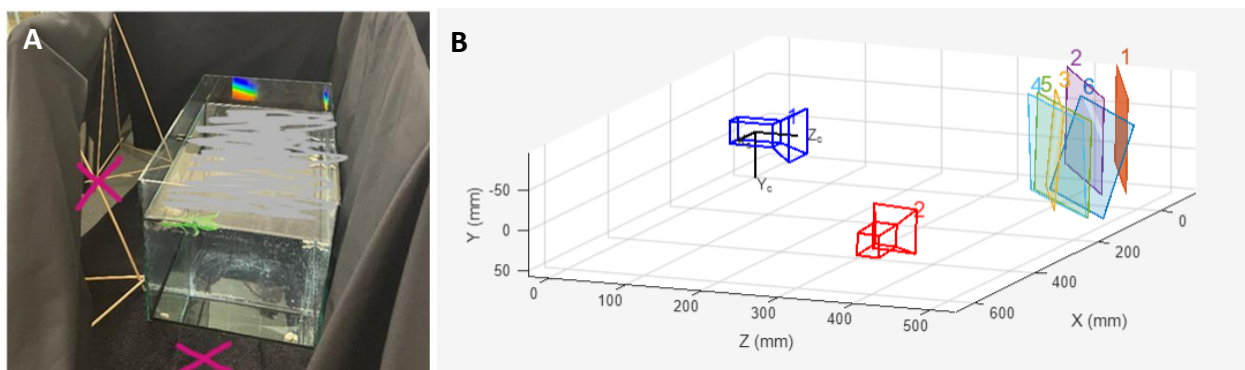
5. Ábra: Levegővétel klasszifikációja videófelvételen. Egy detektált levegővétel (a modell által „1”-ként címkézett szakasz) 99%-os pontossággal (A), illetve egy másik példán pedig a modell jelzi, hogy nem észlelte („0”) a keresett viselkedést (B). Az egyszerűség kedvéért a modellnek a felszíni levegővétel „levegő”-ként volt számon tartva (ékezetek, szóközők és hosszú szavak elkerülése végett).

3.4. Kvantifikáció és klasszifikáció kombinálása

Miután mindkét típusú modell elérte a kívánt pontosságot, mindkettővel ugyanazokat a videókat kellett elemezni, hogy ugyanazon képkockákhoz mind kvantifikációs, mind klasszifikációs adatot tudjunk társítani [lásd Eredmények szekció].

3.5. Háromdimenziós kvantifikáció

Ehhez több, a hangsáv alapján szinkronizált kamerára is szükség volt, hogy ugyanarról a pillanatról több kameraállásból is legyen adatunk a hal pozíciójára vonatkozóan. A kamerák térben egymásra „merőlegesen” (6. ábra) kerültek elhelyezésre, hogy változatosabb szögekre lehessen egyszerre látásunk, így kevésbé hangsúlyossá téve a takarás problémáját.

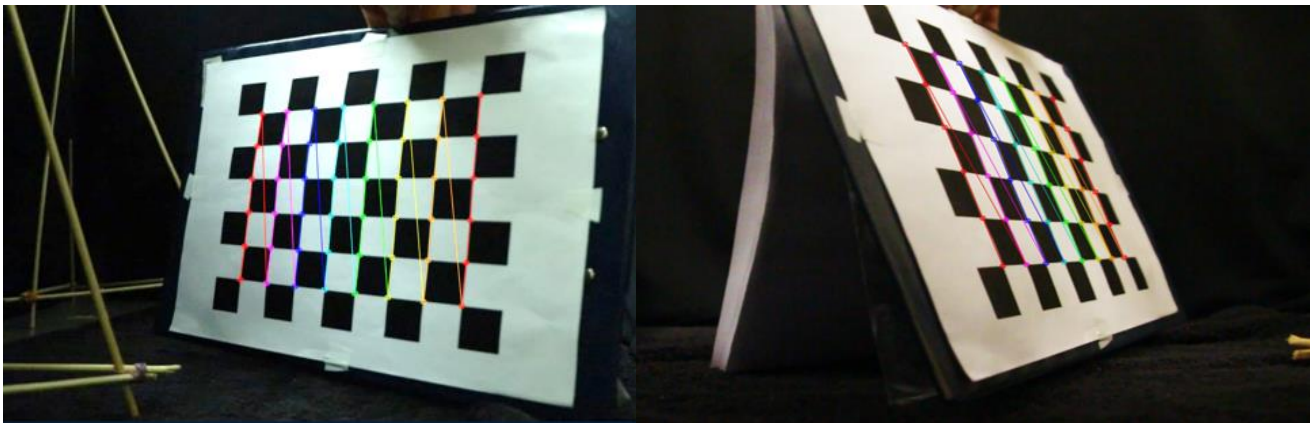


6. Ábra: Háromdimenziós felvételekhez használt beállítások. Az ábrákon a valódi kameraállás (A) és annak a kalibrációs adatokon alapuló MATLAB vizualizációja (B) látható. Utóbbi a Stereo Camera Calibrator App [18] segítségével készült: betöltöttük a sakktáblás adatokat az adott kameraállásra, elvégeztük a kalibrációt, majd vizualizáltuk ez alapján a kamerák elhelyezkedését egy 3D koordináta rendszerben. Ez a kalibrációs eredmény máshol nem került felhasználásra a vizualizáció elkészítésén kívül.

3.5.1 Kalibráció

Továbbá a különböző kamerák felvételeiről begyűjtött 2D-s pozicionális adatok 3D-vé történő összefésüléséhez (háromszögeléséhez) kalibrációra is szükség volt. Ennek során a kamerák külső (pozícióval kapcsolatos) és belső (főként optikai) paraméterei kerülnek kinyerésre. A külső paramétereket (extrinsics) egy 6 dimenziós vektorral (X, Y, Z , α , β , γ) lehet leírni, amely a kamera helyzetét és orientációját határozza meg egy kvázi véletlenszerűen definiált origójú 3D-s koordináta-rendszerben. A belső paraméterek (intrinsics) pedig olyan jellemzőket tartalmaznak, mint például a fókusztávolság, és ezek az adatok főként a lencse által okozott torzítások korrigálásához szükségesek (az ún. undistortion folyamat során) [18].

Tehát a megszokott halas felvételek mellett szükségessé vált ún. kalibrációs felvételek elkészítése is. Utóbbiak során egy ismert paraméterű objektumról kellett videofelvételeket készíteni az eredeti kamerafelállásban, majd annak bizonyos pontjait a különböző kamerák felvételein detektálni. Ebben az esetben az objektum egy papírra nyomtatott és szilárd felületre gondosan felragasztott kockás tábla volt, az ismert paraméterek pedig a négyzetsorok- és oszlopok száma volt. A mezők sarkai kerültek lekövetésre, ez alapján került megállapításra matematikai úton a kamerák külső paraméterei [18, 19, 20] (7. ábra).



7. Ábra: Kamerák kalibrálása 3D felvételekhez. A két használt kamera („szemszög”) sakktábláról készült (kalibrációs) felvételein a mezők sarkait egy speciális algoritmussal (OpenCV [21]) kell detektálni. A tábla sorainak és oszlopainak száma ismert, azaz fix paraméterek. Így a lekövetett mezősarkok helyzetéről begyűjtött adatok alapján lehet matematikai úton egy kalibrációs modellt felállítani.

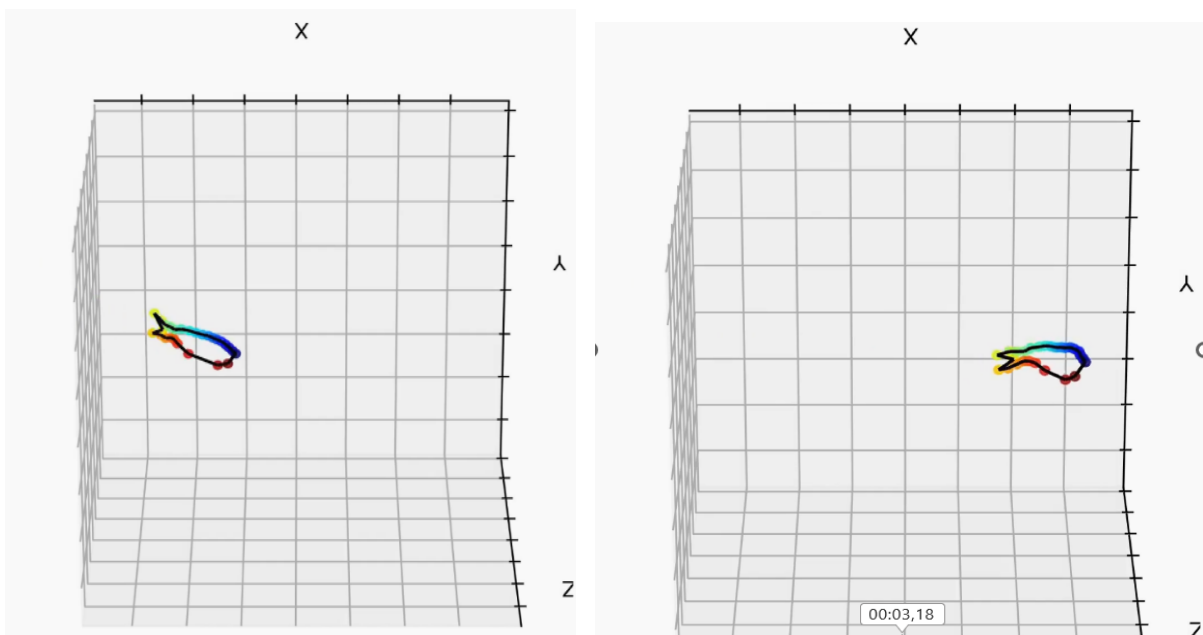
A gyakorlatban először a halas felvételek kerültek leforgatásra, hogy a kamerák pozíciója megfelelően beállításra kerülhessen és az egész akvárium jól látszódjon, majd a kamerák további mozdítása nélkül a halak és az akvárium eltávolítása után, utóbbi helyén egy sakktáblát mozgattunk kamerák előtt.

A kalibrációs adatok és ezáltal felépített kameramodellek pontosságának ellenőrzésére a reprojekciós hiba az egyik leggyakrabban használt mutató: összehasonlítható, hogy a sakktábla mezőinek a kalibráció során meghatározott 3D-s pontjainak vetítése mennyire illeszkedik a kamerák kétdimenziós képén detektált tényleges pontokra. Egy elfogadható kalibráció reprojekciós hibája pixelben mérve nem haladja meg a 3-at, ideális esetben pedig 1 alatt van.

3.5.2 Háromszögelés (2D ->3D)

Amennyiben a reprojekciós hiba elfogadható mértékű (a szóban forgó esetben 0,19 px-es hiba alakult ki), a kalibrációs (mind külső, mind belső) adatok segítségével a kamerák kétdimenziós képére levetített, és ezen lekövetett (hal) testpontokból visszafejthetőek azok eredeti pozíciói a 3D-s térben.

Az összefésült 3D adathalmaz mérete miatt .h5 fájlokban tárolódik, és ellenőrzésképpen vizualizálható (8.ábra).



8. Ábra: Két kamera felvételeinek felhasználása háromszögeléshez. A két kamera felvételein eredetileg külön-külön, 2D-ben lekövetett pozicionális adatok az adott kameraállásra vonatkozó kalibráció alapján rekonstruált eredeti, 3D pozíciója látható különböző pillanatokban.

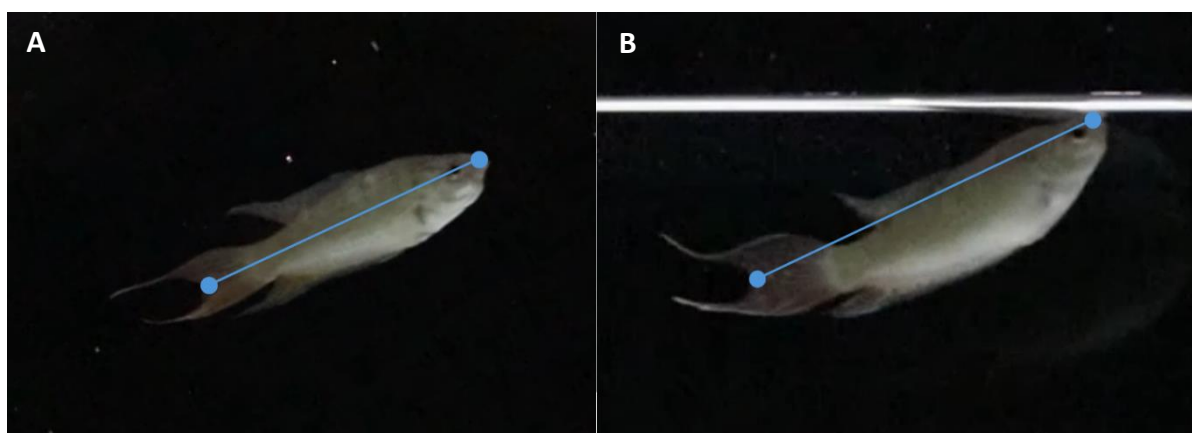
4. EREDMÉNYEK

Minden esetben a fent részletezett módon betanított 27 paradicsomhal testpontot lekövető kvantifikációs- és felszíni levegővétel detektálására alkalmas klasszifikációs modellekkel dolgoztunk, majd ezeket kombináltuk az analízisek során. Így olyan vizsgálatok valósulhattak meg, amelyek eddig csak megfigyelésekre alapulhattak, és adatokkal nem nagyon lehetett őket alátámasztani.

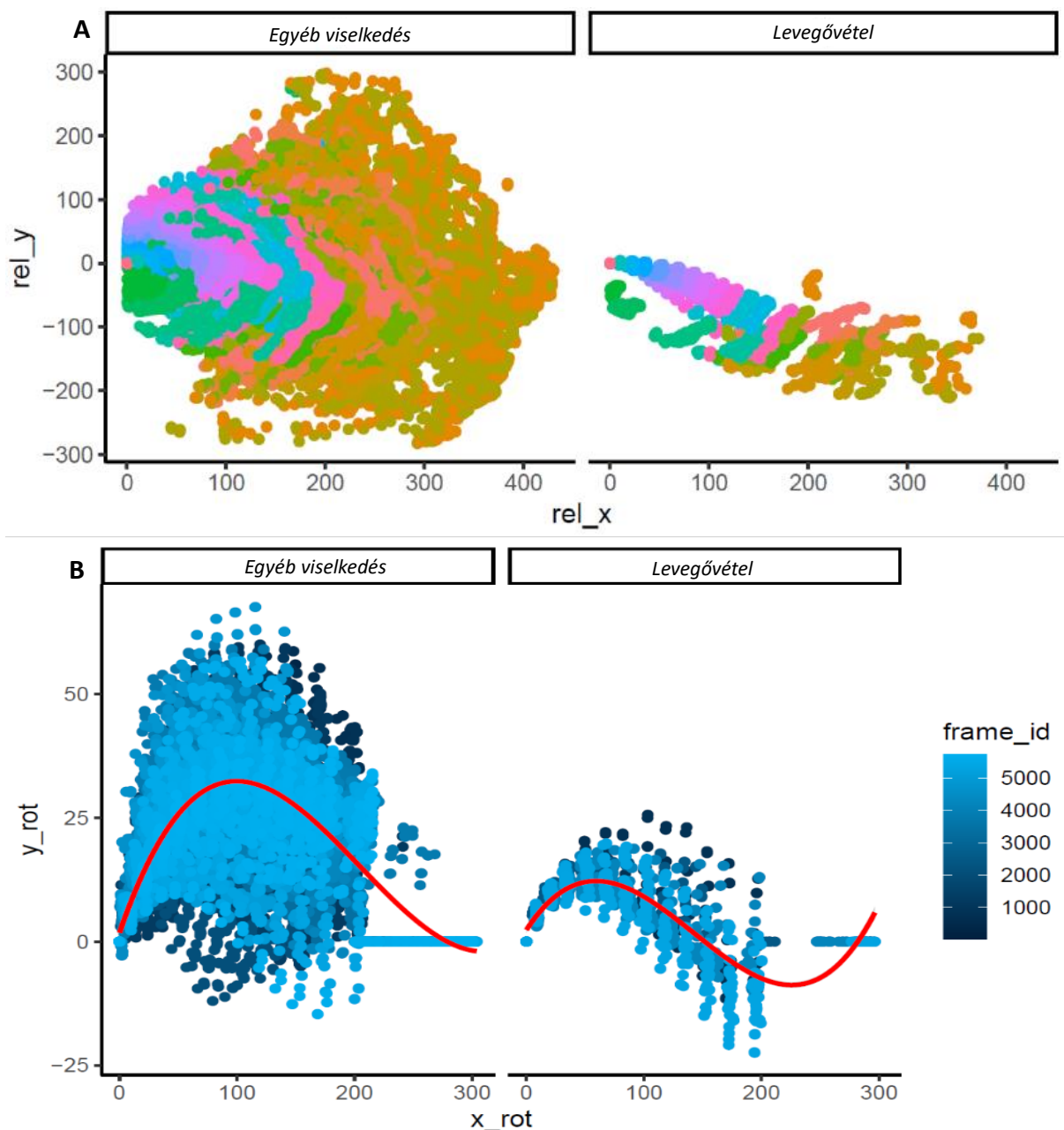
4.2. Felszíni levegővétel részletes elemzése

4.2.1. Hát görbületének vizualizációja

Az eredeti megfigyelésünk (9. ábra) alátámasztására egy egyszerű vizualizáció segítségével összehasonlítottuk a hát görbületét levegővétel során és minden egyéb esetben (10. ábra).



9. Ábra: A paradicsomhalak hátformája levegő vétel során. Felszíni levegővétel során (b) a fark és a hát illesztése közötti szakasz a fent definiált testtengely alá bukik (10B ábra).

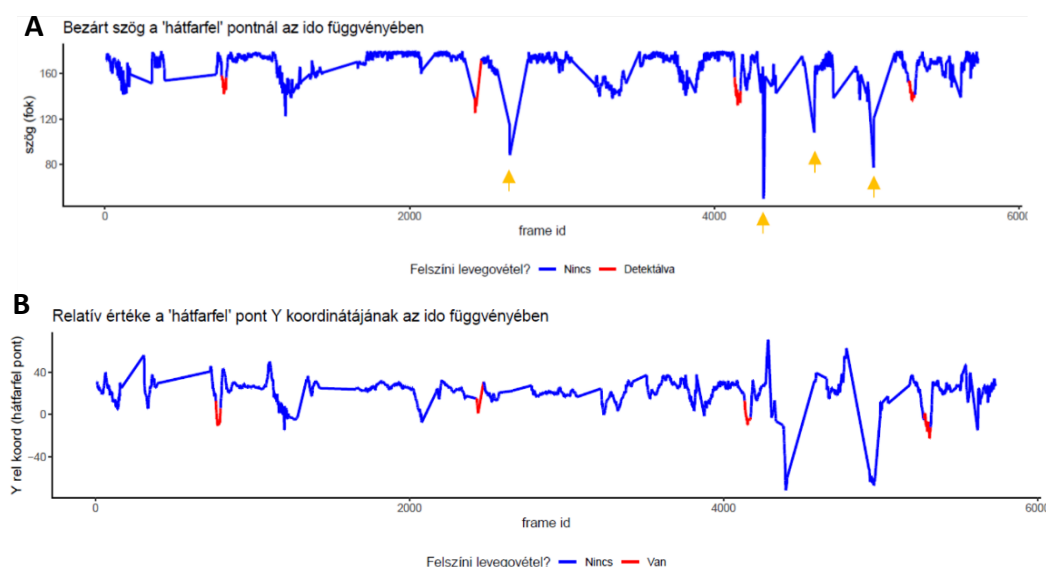


10. Ábra: A paradicsomhalak hátformájának jellemzése levegővétel során. (A) Az orr pont látható minden esetben a koordináta rendszer origójában, és ehhez viszonyítva került az összes többi pont ábrázolásra különböző színekkel (úgy, hogy mindig balra forduljanak.) Ezek az adatok szükségesek tulajdonképpen a következő vizualizáción (17. ábra) végrehajtott koordináta transzformációhoz. (B) A begyűjtött adatok kettéosztva láthatóak aszerint, hogy a klasszifikációs modell „0” vagy „1” értéket rendelt az adott képkockához, és az ehhez tartozó testpozíciós adatok vannak vizualizálva. Szükség volt egy koordináta transzformációra, hogy a testtengely egybeessen az új x tengellyel. Az ábrázolásnál csak a hát oldali pontokat vesszük figyelembe, és az egy képkockához tartozó (hát oldali) pontok ugyanolyan színnel láthatóak. A piros függvény pedig egy, az ezekre ráillesztett polinom (közelítő, harmadfokú függvény) azt mondja meg, hogy „átlagosan” hogy nézett ki a hal háta a két esetben. A következtetés, hogy levegővételnél a hal valóban homorít, amit egzaktabbul mutat az is, hogy a polinom metszi az x tengelyt (ami tulajdonképpen a testtengely).

A látottak egy lehetséges anatómiai indoklása a levegő útjának szabaddá tételének szükségessége a légzés megvalósításához. A folyamat során a levegőnek ugyanis a szájüregen át egy hátrébbi üreghez kell jutnia, ahol megtörténhet a labirintuszszerv segítségével az oxigén kiválasztása. A két üreg között azonban egy „billentyű” [22] helyezkedik el, amelynek ki kell nyílnia a levegő átengedéséhez, és ez a testhelyzet talán segít benne.

4.2.2. Egyéb paraméterek vizsgálata

Ezt követően néhány egyéb paraméter elemzésével tettem kísérletet a látottak további elemzésére. Egyrészt a hát és farok illesztésénél („hátfarfel” pont) keletkező szög vizsgálatával, másrészt az illesztési pont testtengelyhez mért relatív magasságának (y koordináta) elemzésével (11. ábra).



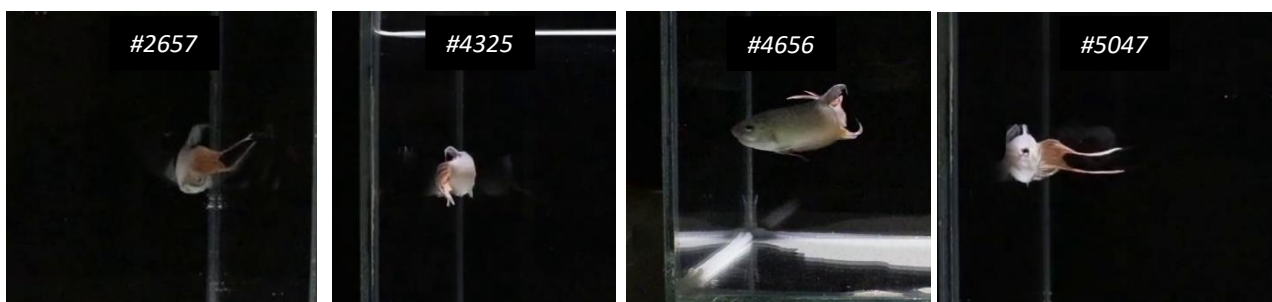
11. Ábra: Az illesztési pontnál („hátfarfel”) keletkező szög nagyságának vizualizációja (A): jól látszik, hogy vannak kiugró csoportok, és a levegővételek is ide tartoznak. Továbbá a viselkedésben mindig megfigyelhető egyfajta „csúcs”: azaz az érték egy ideig csökken, aztán újra nő: ebből az következik, hogy valóban jár valamiféle változással a levegővétel. A váratlanul jobban kiugró, levegő nélküli értékekről (sárga nyilakkal jelölve) pedig kiderült, hogy csak outlierok, amik az optimalizáció során nem kerültek kiszűrésre. (B) Továbbá, mivel a 10B ábrán az új x-tengely (testtengely) alá bukott az illesztett polinom, logikus megvizsgálni, hogy van-e valamiféle mintázat az illesztési pont (hátfarfel) relatív y pozíciójának a váltakozásába (B). Az „A” panelben ábrázolthoz hasonlóan csúcsosodások figyelhetők meg a viselkedés során, illetve néhány outlier miatt okozott inkonzisztencia ugyanott, mint a fenti ábra esetében (ezeket bővebben a 12-13. ábra illusztrálja).

Bár eredményeim szerint a felszíni levegővétellel összefüggésben a „homorításra” utaló (mind az illesztési pont y koordinátájának relatív értékének, mind az itt bezárt szögben megfigyelt) változások konzisztensek, egyéb, levegővételhez nem köthető, sokkal hangsúlyosabb kiugrások is megfigyelhetők. Az inkonzisztencia megértése esetében ezeket közelebbről megvizsgáltuk.

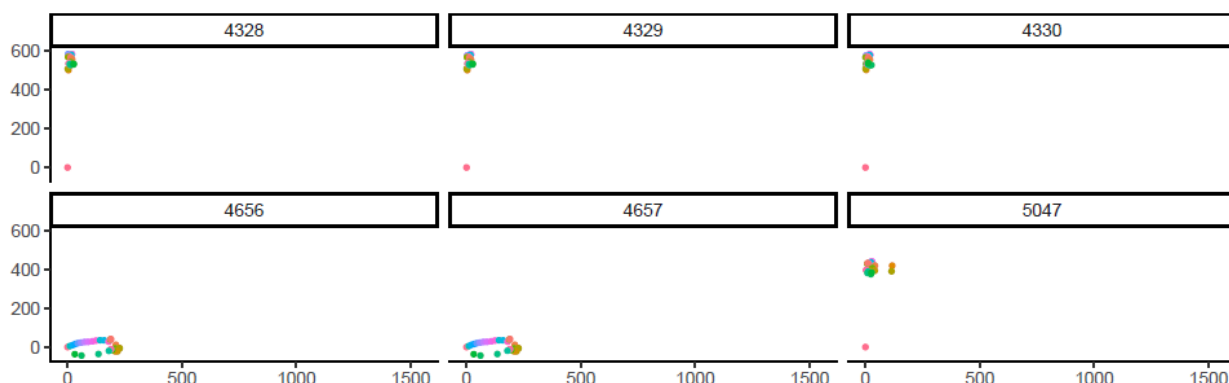
4.2.3. Korrekciók megfontolások kézi elemzéssel

A megfelelő felvételek tüzetes vizsgálata azt igazolta, hogy a fenti vizualizáción megjelenő kilógó adatpontok (outlierek) valójában az *in silico* elemzés kiszűretlen hibáit jelentik (12-13. ábra). Ilyen például, amikor a hal az analízis szempontjából értelmezhetetlen irányba fordult. Ez a megfigyelés kiemeli a 3D-elemzés szükségességét, és felhívja a figyelmet arra, hogy valóban előfordul, hogy a modell hibázik, de ez még az elemzés folyamatában is kijavítható.

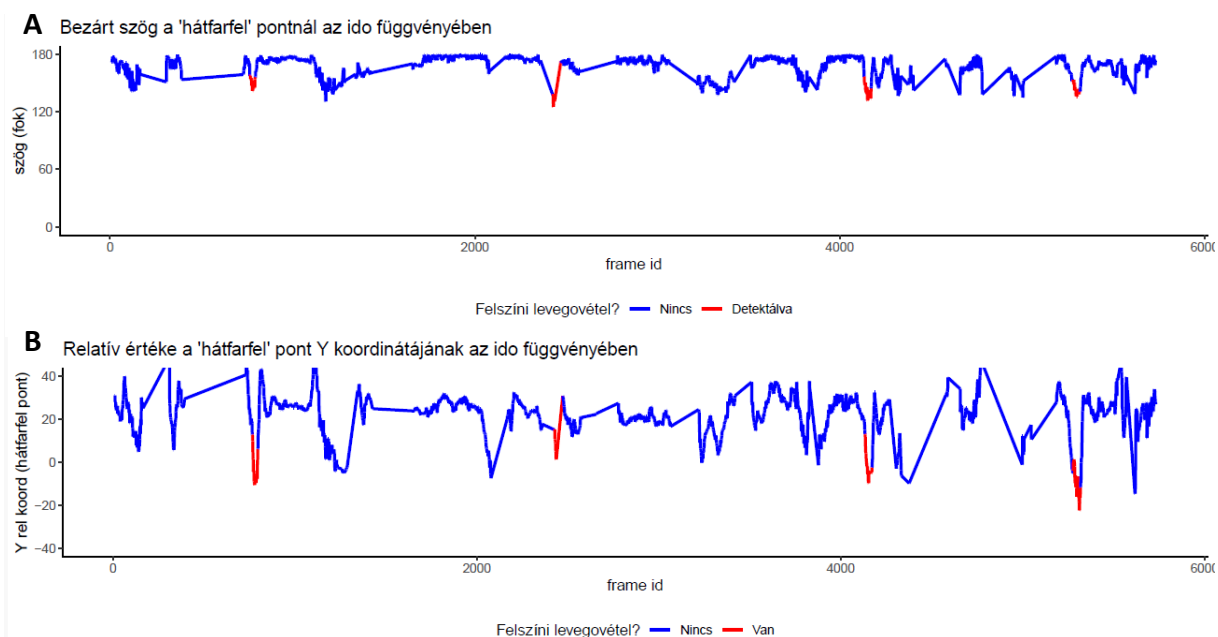
A korrigáció után az immár szűrt adatokon újra lefuttattuk ugyanazt a kódot (14. ábra), így egyértelműbben kirajzolódik a vizsgált jelleg.



12. Ábra: Nehezen elemezhető pozíciók a paradicsomhal viselkedés során. A 2657, 4325, 4656, 5047 számú képkockákat Open CV használatával nyertem ki. Mind a 4 példa különböző „hibás” csúcshoz tartozik: az első két képkockán értelmezhetetlen az adat, mert a hal háttal van nekünk, a második kettőn pedig szintén a hal nem ideális pozíciója miatt a számolások furcsa eredményt adnak. A csúcshoz olyan képkockák is vannak, ahol a kvantifikációs modell hibájából adódik az inkonzisztencia (bővebben a 22. ábra illusztrálja ezeket).



13. Ábra: Inkonzisztens viselkedéselemek vizualizációja. A részletes vizualizáción jól látszik, hogy a 4328-4329-4330 és 5047 esetekben a modell hibázott, a 4656-4657 esetében pedig csak a 2D lekövetés miatt értelmezhetetlen szögbe forduló hal testpontjaiban a várttól eltért eredmények jönnek ki a számolásokra. Utóbbi csak megerősíti a 3D analízis szükségességét. A konklúzió, hogy ezek az esetek mind kiszűrhetőek.



14. Ábra: Az illesztési pontnál keletkező szög vizualizációja, immár a szűrt adatokon (A). Látható, hogy az inkonzisztenciák teljesen eltűntek, és tisztábban kirajzolódik a felszíni levegővétel során látott változás. Továbbá látható az illesztési pont y koordinátájának relatív pozíciójának alakulása az idő függvényében, a szűrt adatokon (B). A csúcsosodások az „A” panelben megfigyelthez hasonlóan itt is hangsúlyosabban kirajzolódnak.

4.2.4. Konklúzió

A fenti adatok azt erősítik meg, hogy valamiféle változás történik a hát görbületével levegővétel során, de ez a jelleg nem feltétlen kizárólagos erre a viselkedésre (tehát máskor is előfordulhat).

5. DISZKUSSZIÓ

A kutatás elsődleges célja non-invazív, videóalapú ML algoritmusok alkalmazása volt a paradicsomhalak viselkedésének részletes elemzéséhez, ezáltal a hagyományosan invazív és hosszadalmas adatbegyűjtési módszertant automatizálva és az emberi hibafaktort csökkentve.

Bár már léteznek tanulmányok hasonló módszertannal más élőlények, akár halfajok esetében is, paradicsomhalakról nem, pedig vizsgálatuk indokolt.

Munkám során sikerült betanításra került egy kvantifikáló és egy klasszifikáló modell is, amelyek megfelelő teljesítményt nyújtottak. Ezt a felszíni levegővétel elemzésénél is demonstráltuk, ahol a modell funkciójának megfelelően azonosította és kvantifikálta a viselkedési mintázatokat. Bár viszonylag kevés adattal dolgoztunk, minden okunk megvan a feltételezésre, hogy a módszer több adattal is ugyanúgy alkalmazható, illetve hogy az eljárás kiterjesztésével az említett hosszú távú célok is megvalósíthatók, beleértve a paradicsomhalak teljes etogramjának elemzését, majd a begyűjtött adatok segítségével további vizsgálatát a viselkedés genetikai hátterének.

Összességében a kitűzött cél sikerült, és a fenti pontosítások alkalmazásával a módszer megbízhatósága és pontossága tovább növelhető.

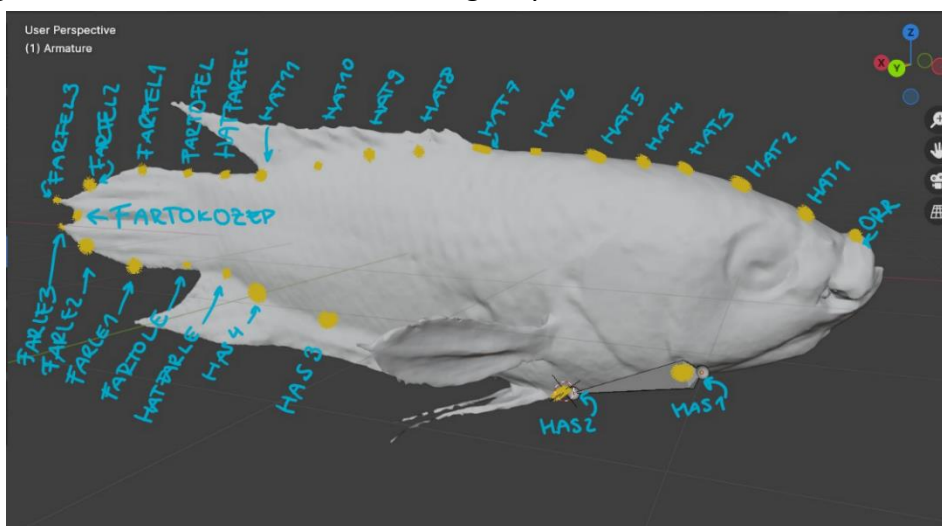
5.1 Kvantifikációs megfontolások

A betanított kvantifikációs modell kis (3.29 px) hibája (az egyértelmű outlierok kivételével) szabad szemmel gyakorlatilag nem látható, illetve a modell teljesítménye további finomhangolása (refinement) és új adatok hozzáadása után kiterjeszthető új felvételekben megjelenő egyéb mozgások detektálására is, akár eltérő háttérrel- és egyéb zajfaktorokkal rendelkező felvételeken is.

5.1.1 Labeling konzisztenciájának javítása

Mivel a modell a felhasználó által felcímkézett adatokon fog tanulni (a betanítási fázis során), a következetesség a címkézés során kulcsfontosságú. Ez a testpontok minél pontosabb definiálásával érhető el: jelen esetben az orr, hát, has, fartő, farok teteje és alja könnyen azonosítható pontok, a többi pedig ezek között, arányos távolságokra került elhelyezésre. A címkézés során bekövetkező, emberi faktor, félreértések miatt bekövetkező hibák elkerülése végett egy 3D paradicsomhal-modellen a kiválasztott testpontok bejelölésre kerültek, így a címkézés során a 3D modell forgatásával az „igazi” adat mellett lehetőség nyílt jobban

megérteni a testpontok közötti interakciót (15. ábra). Továbbá ahelyett, hogy a videó különböző pontjairól csak egyesével kerültek volna kiválasztásra a képkockák, sokszor egy-egy szakasz egyszerre került felcímkézésre. Ez lehetőséget adott arra, hogy az előző képkocka címke-adatait átmásoljuk az adott képkockára, és csak pici változtatásokat végezni rajta, így a pozíció váltakozását, átmeneti állapotait minél precízebben lehet jelölni. Ez a precíz jelölés nagyban segíti a modell későbbi alkalmazhatóságát nyers videóadatokon.

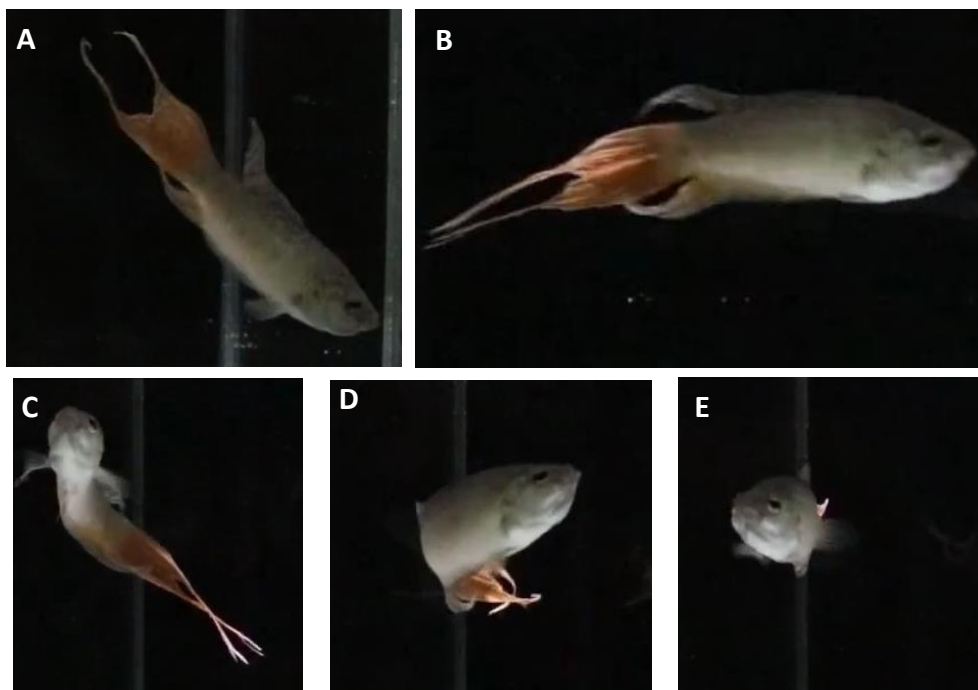


15. Ábra Kritikus testpontok jelölése 3D modellen. A definiált pontok listája és pontos elhelyezkedése: <https://skfb.ly/o9ECI> -ről letöltött 3D paradicsomhal modell, amelyeken a sárga pontok az adott testponton elhelyezett jelölőket ábrázolják.

Bár időtakarékoság szempontjából ésszerűnek tűnhet csak a tervezett analízisnél használatos pontokat felcímkézni, sokszor a „csontváz” (skeleton)-alapú modellkonfigurációk jobban teljesítenek akkor, ha például a test belsejében (szinte a közepén) elhelyezésre kerül egy „extra” pont, még akkor is, ha ennek a pontnak a helyzete a többi testpontból indulva akár kiszámítható is lenne. [23]

Továbbá fontos volt megérteni, hogy nem kerül megkülönböztetésre „könnyen” és „nehezen” felcímkézhető példa (képkocka) a készülő MI modell számára, csak olyan, ami konzisztens címkézés útján meg lett tanítva neki, és olyan, ahol ez nem sikerült. Viszont a felhasználó számára van ilyen (16. ábra) így a betanítás során ez az a pont, ahol az emberi faktor jó és rossz velejárói is belekerülhetnek a rendszerbe. Ennek megoldására hatékony módszernek bizonyult, hogy először mindig (a felhasználó számára) egyszerűbb, azaz olyan képkockák felcímkézésére került sor, amelyeken egyértelműen felismerhető volt a hal minden pontja. A modell ezeken tanulta meg az alapvető mintázatokat. Ezt követően néhány „nehezebb” képkockán került futtatásra, azaz ahol sokszor nem teljesen egyértelmű néhány testpont elhelyezkedése. Ahol a

felhasználó számára elfogadható megoldást talált a modell, az hozzáadásra került a tanítási adathalmazhoz (training datasethez), ahelyett, hogy a felhasználó bizonytalanabb pontosságú adataival történt volna ez.



16. Ábra: Nehezebben címkézhető képkockák. Amikor a hal oldalról látszik (A, B), minden definiált pont egyértelműen kirajzolódik, azonban egyéb orientáció esetében a saját testrészei kitakarhatják egymást (C és D hát pontjai nem látszanak jól, E hal egész második fele kerül takarásba).

Azt is fontos figyelembe venni, hogy mivel egy képkocka csak kis részét teszi ki a mindössze néhány centiméter hosszú hal, az adat maradék része (a háttér) hangsúlyos lesz, hiába tekinthető zajnak. Bár emberi szemmel ezt a „zajt” könnyű kiszűrni, azonban a betanított modellnek szintén meg kell tanulnia kiszűrni ezt a feladat szempontjából semleges információt. Ugyan a labori környezetben előny volt, hogy a felvételekben a háttér viszonylag konstans (fekete) maradt, de még így is előfordulhattak például kopások, repedések a falon. Ha ez a halhoz hasonló színű, az eredményezheti akár a repedésben tévesen „felismert” testpontokat is, hiába tűnik felhasználói szemszögből mindez abszurdnak és érthetetlennek.

További problémát jelenthetnek a tükröződések a modell számára, hiszen – értelemszerűen – saját tükörképe nagyon hasonlít az igazi halra. Ezt a problémát áthidalandó egy olyan megközelítés alkalmazását tervezem, amely alapját egy zebradániókat vizsgáló cikk [24] szolgáltatná, amelyen csak kisebb változásokat kellene rajta végrehajtani (pl. biztosítva a halak számára a levegővétel lehetőségét, és minimalizálva a víz torzító hatásait).

Az emberi tényező szintén befolyásolja az eredményeket, különösen a címkézés során. Bár magának a modellnek nem feltétlenül van szüksége magas minőségű képekre, a címkézés konzisztenciája kritikus fontosságú, és a felhasználó ezt megfelelő minőségű és FPS-sel készült felvételek esetében tudja biztosítani.

5.1.2. Háromdimenziós kvantifikáció szükségessége

Bár az adatok begyűjtése és feldolgozása sikeres volt, ezzel együtt eddigi munkám segített a jelenlegi megközelítés limitációit is beazonosítani, fontos ismereteket adva a jövőbeni munkához.

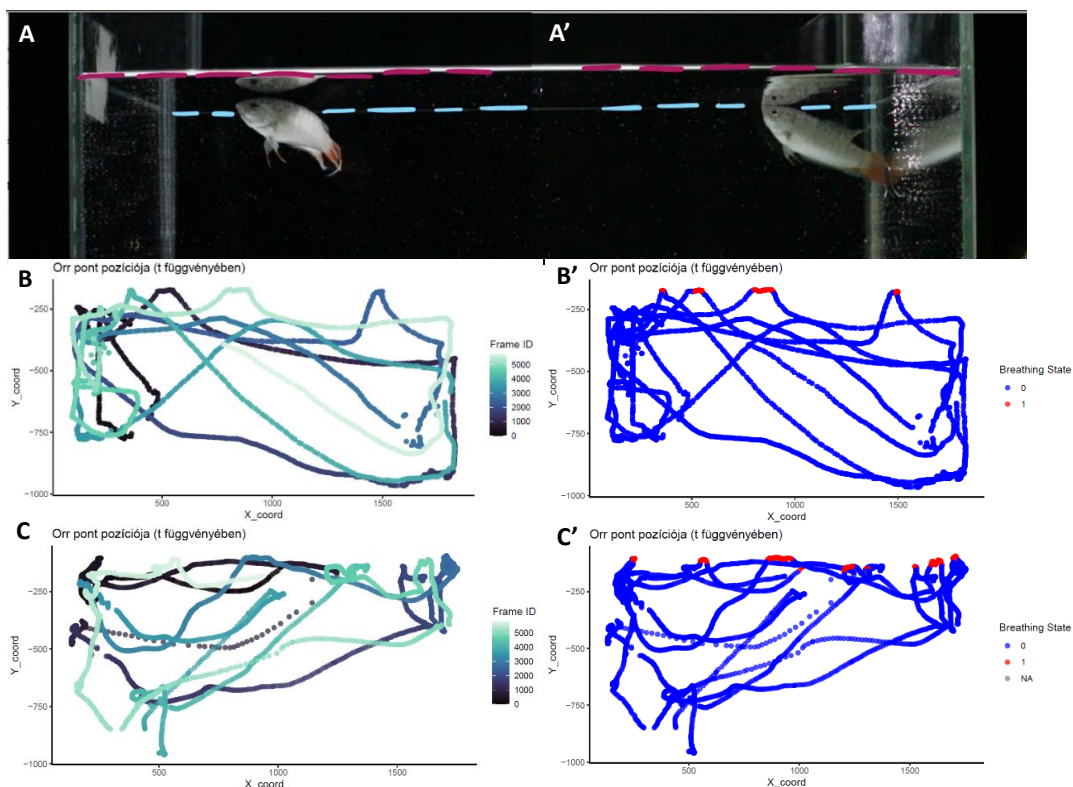
Ilyen következtetés például a 3D kvantifikáció szükségességének felismerése, hiszen nem mindig lehetséges leegyszerűsíteni 2D-re a komplexebb kérdéseket, vagy kikerülni néhány összetettebb problémát (pl. takarások, főleg többállatos elemzéseknél) [29].

Ugyanakkor megemlíteném a leírt módszertanban a 3D címkézés hiánya: különösen az általunk használt merőleges kamerabeállítás hátránya, hogy a túl különböző képek könnyebben felboríthatják a címkézés konzisztenciáját, ami rontja a modell pontosságát. A jövőben, nagyobb adatmennyiséggel dolgozva, szükséges lesz a 3D-s címkézés megvalósítása. Ez azt jelentené, hogy az ugyanazon pillanatra vonatkozó felvételeket egyszerre címkéznénk fel, hiszen valójában ugyanarra a 3D-s objektumra vonatkoznak, csak a videó definíciószerűen 2D-re egyszerűsíti le azt [25].

5.2. Klasszifikáció

5.2.1. A klasszifikáció szükségessége

Kezdetben azt feltételeztük, hogy a viselkedés egyszerű természete és (első ránézésre) könnyű definiálhatósága miatt pusztán a kvantifikációs adatokból a viselkedés minden esetben meghatározható. Azonban az alábbi vizualizációk jól illusztrálják, hogy ez akár ennyire egyszerű viselkedéseknél sem teljesül mindig, indokolva a kifejezetten klasszifikációs modellek használatát. (17.ábra)



17. Ábra: két különböző képkocka egymás mellett, a kék és lila szaggatott vonalak között jól látszik a vízfelszín tényleges területe, és ez emiatt keletkezik „szürke zóna”. Hiába van magasabb értéke az orr pont y értéke az „A” példának, ezen nem történik levegővétel, az „A’ ” példán pedig, ahol az orr pont y értéke alacsonyabban van, megfigyelhető a keresett viselkedés. Ugyanezt a problémát adatokon is vizualizáltam. Bár előfordul, hogy az orr y értékével konzisztens a levegővételek előfordulása (B és B’), előfordulhat, hogy ugyanazon y érték néha levegővételt jelent, néha nem (C és C’).

5.2.2. A választott klasszifikációs stratégiák indoklása

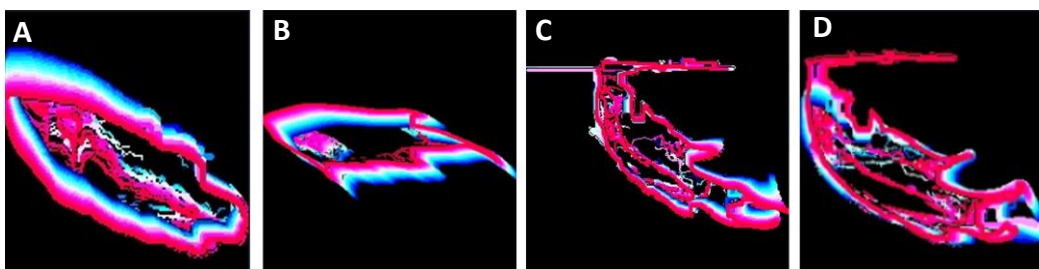
Azt fontos megemlíteni, hogy többféle klasszifikációs modellstruktúra is alkalmas lehet különféle kutatási kérdések megválaszolására: az elemzés alapja lehet nyers képi adat (a videókból), illetve a már azok feldolgozásával létrehozott pozicionális adat is. A képi adatokkal dolgozó klasszifikációs modellek a jellegzetes sziluettű viselkedések (pl. felszíni levegővétel) detektálásában alkalmazhatóak jól, illetve a klasszifikáció ez esetben teljesen független a kvantifikációtól [26].

A pozicionális adatokat elemző modellek esetében bár a gyorsabb feldolgozási folyamat egyértelmű előny, kiemelt szerepet kell kap a kvantifikálásnál a megfelelő testpontok kiválasztása, azaz, hogy biztosan lekövetésre kerüljenek a viselkedés alakításában meghatározó módon részt vevő testpontok is. Tehát kvantitatív adatokat használó analízis esetén a klasszifikáció nem lehet független a kvantifikációtól, sőt, előbbi nagyon is jelen van potenciális hibafaktorként. [26]

A klasszifikációs modellek tanulási folyamata megoldható a felügyelt (supervised) és a felügyelet nélküli (unsupervised) módon is. Előbbinél a betanítás során a modell számára a felhasználó biztosítja a „példákat” a detektálandó viselkedésekre, utóbbinál pedig ún. clustering történik, azaz a modell keres ismétlődő mintázatokat a kapott adathalmazban és ez alapján csoportosítja őket. Utóbbinál az emberi (hiba)faktor teljes kivonása lenne megvalósítható, viszont előbbi előnyösebb, hogyha kifejezetten egy adott viselkedést keresünk [27].

A fentebb részletezett metodológiai megfontolások miatt a felszíni levegővétel detektálásához felügyelt és képi-alapú modell került alkalmazásra (jellegzetes sziluett, pontosan meghatározott viselkedés, független kvantifikáció és klasszifikáció).

Több egyéb algoritmus tesztelése után azért választottam a LabGym algoritmust, mert kiderült, hogy a kontrasztkülönbségek elemzésével egyszerűsíti le a kapott adatokat [11]. Mivel a vízfelszínen máshogy törik meg a fény (szinte egy fehér csíkként látszik), be fog kerülni a szűrt adathalmazba és részt fog venni a modell betanításában is (18. ábra). Ezáltal a modellünk nagy biztonsággal ismerheti fel azokat az eseteket, ahol a hal (sziluettje) érintkezik ezzel a vonallal. Ezek mellett lehetőség nyílik arra is, hogy a modellnek egyéb paramétereket megtanítsunk, így a felszín többi síkjában történő levegővételeket is sikeresen azonosította. Az objektivitás kedvéért fontos azonban megjegyezni, hogy az MI működésének „fekete doboz” jellege miatt nem lehet teljes bizonyossággal meghatározni, hogy tényleg a leírt mechanizmus (itt a kontúrok összeolvadása) hozzájárult-e a modell sikeréhez, így a fentebb leírtakra csak a modell struktúrájából kiindulva következtetek [28].



18. Ábra: A LabGym klasszifikációs modell) által feldolgozott példák. Egyes esetekben csak úszás (A, B) közben látható a sziluett, máshol felszíni levegővétel is megfigyelhető (C, D). Utóbbi esetben jól látszik a viselkedés jellegzetes sziluettje mellett az is, hogy a vízfelszín valóban belekerülhet a tanulási adathalmazba.

Továbbá az adott magatartásformákhoz rendelt szakaszok időtartamával kísérletezni kellett, hiszen a levegővétel általában az úszási folyamat egy része, így nehezen húzható meg egyértelműen az a határ, hogy honnantól kezdődik ez a konkrét viselkedés: csak a viselkedés lényegi, ismétlődő szakaszát tanulja meg, anélkül, hogy a környező, kevésbé releváns elemeket is beépítené a mintázatba, ami esetleg nem mindig jelenik meg. A jövőben emiatt érdemes lenne felügyelet nélküli modellekkel is kísérletezni, ezáltal teljesen kivonva az emberi faktort, szubjektivitást az említett szürke zónára vonatkozóan is [26].

KÖSZÖNETNYILVÁNÍTÁS

Mindenekelőtt hálás köszönettel tartozom a témavezetőmnek, Dr. Varga Máténak a támogatásáért és útmutatásáért, és a biológia tanáromnak, Nádori Gergelynek a további építő jellegű kritikáért. A projekt nem jöhetett volna létre az ELTE Genetika Tanszék Halgenetika Kutatócsoportjának összessége nélkül, kiemelve Czimer Dávidot és Gyulai Zsoltot, akik a videofelvételek elkészítésében voltak segítségemre.

FELHASZNÁLT IRODALOM

- [1] Szabó, N., Fodor, E., Varga, Z., Tarján-Rácz, A., Szabó, K., Miklósi, Á., & Varga, M., "The paradise fish, an advanced animal model for behavioral genetics and evolutionary developmental biology," *Journal of Experimental Zoology Part B: Molecular and Developmental Evolution*, vol. 342, pp. 189-199, 2024.
- [2] V. Csányi and A. Dóka, "Learning interactions between prey and predator fish," *Marine Behaviour and Physiology*, vol. 23, no. 1-4, pp. 63-78, 1993.
- [3] Vilmos Csányi és R. Gerlai, „Open-Field Behavior and the Behavior-Genetic Analysis of the Paradise Fish (*Macropodus opercularis*)” *J. Comp. Psych.*, sz. 102 (1988): 226–36
- [4] A. Mathis and M. W. Mathis, "Deep learning tools for the measurement of animal behavior in neuroscience," *Current Opinion in Neurobiology*, vol. 60, no. ISSN 0959-4388, pp. 1-11, 2020.
- [5] Fazzari, E., Romano, D., Falchi, F., & Stefanini, C. (2023). Animal Behavior Analysis Methods Using Deep Learning: A Survey. arXiv. <https://doi.org/10.48550/arXiv.2405.14002>
- [6] Koçkaya, M., Isparta, S., Reinhardt, P.R. et al. Tracking puppy development: automated analysis and qualitative behavioral assessment in repeated open field tests. *Vet Res Commun* 49, 35 (2025).
- [7] Chen, CH., Lin, YC., Wang, SH. et al. An automatic system for recognizing fly courtship patterns via an image processing method. *Behav Brain Funct* 20, 5 (2024).
- [8] S. Lee, B. Waugh, G. O'Dell, X. Zhao, W. -S. Yoo and D. H. Kim, "Predicting Fruit Fly Behaviour using TOLC device and DeepLabCut," *2021 IEEE 21st International Conference on Bioinformatics and Bioengineering (BIBE)*, Kragujevac, Serbia, pp. 1-6 (2021).
- [9] <https://danio.elte.hu/index.html>.
- [10] Nath, T., Mathis, A., Chen, A.C. et al., "Using DeepLabCut for 3D markerless pose estimation across species and behaviors," *Nat Protoc*, vol. 14, pp. 2152-2176, 2019.
- [11] Yujia Hu, Carrie R. Ferrario, Alexander D. Maitland, Rita B. Ionides, Anjesh Ghimire, Brendon Watson, Kenichi Iwasaki, Hope White, Yitao Xi, Jie Zhou, Bing Ye, "LabGym:

Quantification of user-defined animal behaviors using learning-based holistic assessment," Cell Reports Methods, Vols. 3, Issue 3, 2023.

[12] Jouary, A., Silva, P. T. M., Laborde, A., Mata, J. M., Marques, J. C., Collins, E. M. D., Peterson, R. T., Machens, C. K., & Orger, M. B. (2024). Megabouts: A flexible pipeline for zebrafish locomotion analysis. bioRxiv. <https://doi.org/10.1101/2024.09.14.613078>

[13] Pereira, T.D., Tabris, N., Matsliah, A. et al. SLEAP: A deep learning system for multi-animal pose tracking. Nat Methods 19, 486–495 (2022).

[14-15] Aliferis, C., Simon, G. (2024). Overfitting, Underfitting and General Model Overconfidence and Under-Performance Pitfalls and Best Practices in Machine Learning and AI. In: Simon, G.J., Aliferis, C. (eds) Artificial Intelligence and Machine Learning in Health Care and Medical Sciences. Health Informatics. Springer, Cham.

[16] Shorten, C., Khoshgoftaar, T.M. A survey on Image Data Augmentation for Deep Learning. J Big Data 6, 60 (2019).

[17] Deisenroth, Marc & Ohlsson, Henrik. (2011). A general perspective on Gaussian filtering and smoothing: Explaining current and deriving new algorithms. 1807 - 1812. 10.1109/ACC.2011.5990871.

[18] MathWorks. (2024). Using the Stereo Camera Calibrator App. (<https://www.mathworks.com/help/vision/ug/using-the-stereo-camera-calibrator-app.html>)

[19] Zhang, Z. "A Flexible New Technique for Camera Calibration". IEEE Transactions on Pattern Analysis and Machine Intelligence. 22, no. 11 (November 2000): 1330–34.

[20] Heikkila, J., and O. Silven. "A Four-step Camera Calibration Procedure with Implicit Image Correction." In Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1106–12. San Juan, Puerto Rico: IEEE Comput. Soc, 1997.

[21] OpenCV Team. (2024). OpenCV: Open Source Computer Vision Library. (<https://opencv.org/>)

[22] H. M. Peters, "On the mechanism of air ventilaton in anabantoids (Pisces: Teleostei)," Zoomorphologie, vol. 89, pp. 93-123, 1978.

[23] <https://deeplabcut.github.io/DeepLabCut/docs/recipes/nn.html>

- [24] L. O'Shaughnessy, T. Izawa, I. Masai, J. W. Shaevitz, and G. J. Stephens, "Dynamics of dominance in interacting zebrafish," *PRX Life*, no. 3. 10.1103/PRXLife.2.043006., 2024.
- [25] Y. Han, K. Chen, Y. Wang, W. Liu, X. Wang, J. Liao, Y. Huang, C. Han, K. Huang, J. Zhang, S. Cai, Z. Wang, Y. Wu, G. Gao, N. Wang, J. Li, Y. Song, J. Li, G. Wang, L. Wang, Y. Zhang, and P. Wei, "Social Behavior Atlas: A computational framework for tracking and mapping 3D close interactions of free-moving animals," *bioRxiv*, no. 2023.03.05.531235., 2023.
- [26] M. Pérez and C. Toler-Franklin, "CNN-Based Action Recognition and Pose Estimation for Classifying Animal Behavior from Videos: A Survey," *arXiv*, no. 2301.06187v1, 2023.
- [27] M. Sur, J. Hall, J. Brandt, M. Astell, S. Poessel and T. Katzner, "Supervised versus unsupervised approaches to classification of accelerometry data," *Ecology and Evolution*, no. 13. 10.1002/ece3.10035., 2023.
- [28] Buhrmester, V., Münch, D., & Arens, M. (2019). Analysis of Explainers of Black Box Deep Neural Networks for Computer Vision: A Survey. *arXiv*.
<https://doi.org/10.48550/arXiv.1911.12116>
- [29] T. Fukunaga, S. Kubota, S. Oda and W. Iwasaki, "GroupTracker: Video tracking system for multiple animals under severe occlusion," *Computational Biology and Chemistry*, vol. 57, pp. 39-45, 2015.