

Halfajták és méreteik meghatározása sztereó digitális képfeldolgozással

Fish size and type estimation using stereo image processing

BARTHA Áron^{1,2}, BARTHA Csaba¹, TÓKOS Attila¹, CÎRCIMARU Gabriela³

¹ Research-Development Institute for Environmental Protection Technologies and Equipment – I.C.P.E. Bistrița S.A., Str. Parcului, Nr. 7, 420035, Bistrița, BN-RO, Fax: +40 263 210938, <https://www.icpebn.ro>, csaba.bartha@icpebn.ro

² Babeş-Bolyai Tudományegyetem, Kolozsvár, <https://www.ubbcluj.ro>

³ INC DIE ICPE-CA, Splaiul Unirii nr. 313, sector 3. Bucureşti www.icpe-ca.ro, gabrielaoprina@yahoo.com

Abstract

Fish size and distance from the camera are estimated by digitally processing stereo images. Calibration provides the camera geometry and physical scale. Differences between the image positions of corresponding points yield three-dimensional coordinates. A pretrained instance-segmentation model selects the surface points of each fish for size and distance estimation; a server-oriented software prototype performs the geometric evaluation.

Keywords: digital image processing, stereo vision, camera calibration, disparity, fish segmentation

Kivonat

Sztereókamerás felvételek digitális feldolgozásával becsüljük a halak méretét és kamerától mért távolságát. A kalibráció a kamerák geometriáját és a fizikai méretarányt adja meg. A képeken azonosított pontok helyzetkülönbségéből térbeli koordináták számíthatók. Az előtanított példányszegmentáló modell halanként kijelöli a méret- és távolságbecsléshez használt felületi pontokat; a geometriai kiértékelést szerverre tervezett prototípus végzi.

Kulcsszavak: digitális képfeldolgozás, sztereó látás, kamerakalibráció, diszparitás, halszegmentálás

1. BEVEZETÉS

A halak méretének pontos ismeretére számos kutatási és gyakorlati alkalmazásban szükség van. A testméret változásából következtetni lehet az egyedek növekedésére, fejlődésére és az állomány állapotára, ugyanakkor a méretadatok felhasználhatók az egyedszám, illetve a biomassza becsléséhez is. Hagyományos mérés esetén azonban a halakat ki kell fogni, majd egyenként meg kell mérni, ami nagyobb állomány esetén időigényes, rendszeres mérésre nehezen alkalmazható, továbbá az állatok számára is zavaró beavatkozást jelent. Ezért előnyös lenne egy olyan érintésmentes eljárás alkalmazása, amely a vízben szabadon mozgó halak méretét képfeldolgozással, automatikusan képes meghatározni.

A halak automatikus felismerésének és méretük meghatározásának jelentősége túlmutat az állományok egyszerű megfigyelésén. A felszíni vizek ipari, ivóvízellátási vagy energetikai célú hasznosításához kapcsolódó műtárgyak megváltoztathatják a halak természetes élőhelyét és vándorlási útvonalait. A vízkivételeknél és vízerőműveknél ezért különböző mechanikai vagy viselkedési elven működő halvédelmi megoldások alkalmazhatók. A mechanikai rácsok sűrűségének növelése ugyan csökkentheti a halak veszélyes területekre történő bejutását, ezzel együtt azonban növeli a hidraulikai ellenállást, valamint az eltömődés és a karbantartás problémáját. Alternatívát jelentenek a fény-, hang-, elektromos vagy hidrodinamikai ingereket alkalmazó viselkedési sorompók, amelyekkel a halak fizikai akadály nélkül terelhetők. Az ilyen ingerek hatása ugyanakkor függhet többek között a hal fajától, méretétől és fejlettségi állapotától [1-5].

Ebből következően egy automatikus halfelismerő és -méretező rendszer egy intelligens halvédelmi rendszer számára is hasznos kiegészítő információt szolgáltat. Amennyiben például egy vízkivételi műtárgy előtt alkalmazott mechanikai rács nyílásmérete ismert, a felismert hal becsült mérete alapján megállapítható,

hogy az egyed várhatóan képes-e átjutni a rácson. Ha mérete miatt erre eleve nem képes, az aktív viselkedési sorompó működtetése szükségtelen lehet, ezáltal csökkenthető a fény-, hang- vagy más inger előállításához szükséges energiafelhasználás. Az igény szerinti működtetés egy további lehetséges előnye, hogy a halak nincsenek folyamatosan ugyanannak az ingernek kitéve. Ez csökkentheti az ingerhez történő hozzászokás, vagyis a habituáció kialakulásának lehetőségét, amely tartós működtetés esetén ronthatja egy viselkedési sorompó hatékonyságát. Az automatikus méretmeghatározás tehát nemcsak megfigyelési célokat szolgálhat, hanem egy nagyobb automatizált rendszer döntéshozatalának egyik bemenő adata is lehet. [1-5]

Egy hagyományos, egyetlen kamerával készített képen azonban a hal képen látható mérete önmagában nem árulja el a testhosszát: egy közeli, kisebb és egy távolabbi, nagyobb példány ugyanakkorának látszhat.

A probléma egyik lehetséges megoldását két, egymáshoz képest ismert helyzetben rögzített kamera együttes alkalmazása jelenti. A sztereókamerás elrendezés működési elve az emberi térlátáshoz hasonló: ugyanaz a térbeli pont a két, egymástól meghatározott távolságra elhelyezett kamera képen kissé eltérő helyen jelenik meg. Ha a kamerák geometriai tulajdonságai és egymáshoz viszonyított helyzete kalibrációval meghatározható, akkor a két képen megfigyelhető eltérésből kiszámítható a pont térbeli helyzete és kamerától mért távolsága.

Ezekre való tekintettel dolgozatunk célja egy olyan képfeldolgozó eljárás kidolgozása, amely sztereókamerás felvételek alapján automatikusan felismeri a képen található halakat, elkülöníti őket a környezetüktől, majd a két kamerakép és a kalibrációs adatok felhasználásával megbecsüli azok térbeli helyzetét, kamerától mért távolságát és fizikai méretét.

2. A MEGOLDANDÓ PROBLÉMA

A feladat a halak képi elkülönítése, fizikai méretük és kamerától mért távolságuk meghatározása. Ehhez az egyedekhez külön kijelölést kell rendelni, majd ugyanazt a halat a két kameraképen összepárosítani. A halfajok pontos azonosítása szintén cél: az elkülönített egyedeket a megfelelő fajhoz kell rendelni. A méret- és helyzetadatok egy halvédelmi rendszer vezérlését is támogathatják.

3. A SZTEREÓ DIGITÁLIS KÉPFELDOLGOZÁS MENETE

A Python-program először eltárolja a kalibrációból származó kamerageometriát. A későbbi képpárokat ezzel igazítja össze. Az OpenCV a képpontok térbeli helyét számítja, a felismerő modell pedig kijelöli a halakhoz tartozó képpontokat. A NumPy ezekből becsüli az egyes halak kiterjedését. A kódrészletekben cv2 az OpenCV, np a NumPy rövid neve. [9]

4. A KAMERAKÉPTŐL A TÉRBELI PONTIG

4.1. Kalibráció és fizikai méretarány

A torzításmentes kameramoddell a térbeli (X; Y; Z) ponthoz az alábbi képi helyet rendeli [6]:

$$u = f_x \frac{X}{Z} + c_x, \quad v = f_y \frac{Y}{Z} + c_y \quad (1)$$

Az f_x , f_y képpontban kifejezett gyújtótávolság és a (c_x ; c_y) főpont a K kameramátrix részei. A kalibráció a sakktábla ismert, milliméteres sarokkoordinátáit (object_points) kapcsolja a két kép sarokpontjaihoz (image_points_left, image_points_right). Az image_size a képfelbontás.

1. kódrészlet. Kalibráció és a kamerák egymáshoz viszonyított helyzete (calibration.py)

```
rms_left, K1, D1, _, _ = cv2.calibrateCamera(
    object_points, image_points_left, image_size, None, None)
rms_right, K2, D2, _, _ = cv2.calibrateCamera(
    object_points, image_points_right, image_size, None, None)
criteria = (cv2.TERM_CRITERIA_EPS + cv2.TERM_CRITERIA_MAX_ITER, 200, 1e-7)
rms_stereo, K1, D1, K2, D2, R, T, E, F = cv2.stereoCalibrate(
    object_points, image_points_left, image_points_right,
    K1, D1, K2, D2, image_size,
    criteria=criteria, flags=cv2.CALIB_FIX_INTRINSIC)
```

A `calibrateCamera` a kameránkénti K mátrixot és D torzítási együtthatókat becsüli. A `stereoCalibrate` ezeket rögzítve határozza meg az R forgatást és a T eltolást. A T hossza a milliméteres B bázistávolság. K , D , R és T a képek összeigazításának és a későbbi térbeli visszaszámításnak a bemenetei.



1. ábra. A sakktáblás kalibráció két nézete, az azonosított belső sarokpontokkal

Az ismert táblapózú Blender-próbában $f_x = 2666,577$ és $f_y = 2666,117$ képpont, a főpont (959,512; 539,511) volt. A sarokpontok visszavetítési RMS-eltérése 0,177 képpont.

4.2. Rektifikáció és képpontpárosítás

A rektifikáció olyan virtuális kameranézeteket állít elő, amelyekben az összetartozó pontok ugyanabba a képsorba kerülnek. A `stereoRectify` K , D , R és T alapján adja meg a képekhez tartozó R_1 , R_2 forgatásokat, P_1 , P_2 vetítési mátrixokat és a visszaszámításhoz szükséges Q mátrixot. Az `initUndistortRectifyMap` kijelöli az átmintavételezés helyeit, a `remap` pedig elkészíti a korrigált képeket [6].

A vízszintes képi eltolódás, vagyis a diszparitás $d = u_L - u_R$. A `StereoSGBM` a rektifikált képek szürkeárnyaltos, helyi kontrasztjavítással előkészített változatain (`gray_left`, `gray_right`) hasonlítja össze a kis képrészleteket.

2. kódrészlet. A halas futás diszparitásszámítása (`stereo.py`)

```
matcher = cv2.StereoSGBM_create(
    minDisparity=0, numDisparities=640, blockSize=5,
    P1=8 * 5 * 5, P2=32 * 5 * 5,
    disp12MaxDiff=1, preFilterCap=31, uniquenessRatio=8,
    speckleWindowSize=100, speckleRange=2,
    mode=cv2.STEREO_SGBM_MODE_SGBM_3WAY,
)
raw_left = matcher.compute(gray_left, gray_right)
disparity = raw_left.astype(np.float32) / 16.0
```

A `compute` 5×5 képpontos környezeteket hasonlít össze a nullától induló, 640 képpontos eltolódási tartományban. P_1 és P_2 a szomszédos diszparitások indokolatlan ugrásait mérséklő büntetőparaméterek; a további beállítások a bizonytalan és elszigetelt találatokat szűrik [7]. A nyers eredmény 16-tal osztva adja a képpontban kifejezett `disparity` térképet: a 384 nyers érték például 24 képpontos eltolódás.

4.3. A diszparitástól a térbeli koordinátákig

Párhuzamos, rektifikált kameráknál, azonos főpontonál $u_L = fX/Z + c_x$ és $u_R = f(X - B)/Z + c_x$. A két vetület különbségéből kiesik X és c_x , így:

$$d = \frac{fB}{Z}, \quad Z = \frac{fB}{d}, \quad X = \frac{(u - c_x)Z}{f}, \quad Y = \frac{(v - c_y)Z}{f} \quad (2)$$

A B bázistávolság milliméteres egysége adja a térbeli koordináták egységét. A bal kamerától mért távolság $r = \sqrt{X^2 + Y^2 + Z^2}$. A `reprojectImageTo3D` a Q mátrixszal minden (u ; v ; d ; 1) adatsorból homogén térbeli koordinátákat képez, majd a negyedik komponenssel osztva előállítja X , Y és Z értékét [6].

3. kódrészlet. Térbeli koordináták és érvényességi kijelölés (stereo.py)

```
xyz = cv2.reprojectImageTo3D(
    disparity.astype(np.float32), calibration.Q)
valid = np.isfinite(disparity) & np.isfinite(xyz).all(axis=2)
valid &= disparity > (config.min_disparity + 0.25)
valid &= xyz[:, :, 2] >= config.min_depth_mm
valid &= xyz[:, :, 2] <= config.max_depth_mm
```

Az `xyz[v, u]` a bal képponthez tartozó térbeli pontot tárolja: X jobbra, Y lefelé, Z előre mutat. A `valid` a véges és megengedett mélységű pontokat jelöli. Ezek `result.xyz_mm` és `result.valid_mask` néven kerülnek a halankénti kiértékeléshez.

4.4. Számítási példa

$B = 1000$ mm, $f = 50 \cdot 1920 / 36 \approx 2666,667$ képpont és $d \approx 533,333$ képpont esetén $Z \approx 5000$ mm. A (960; 540) főpont és a bal képen (1200; 540) pont mellett $X \approx 450$ mm, $Y = 0$, $r \approx 5020,2$ mm. Ugyanebben a mélységben 100 képpontnyi vízszintes kiterjedés 187,5 mm-nek felel meg.

5. A TÉRBELI PONTOKTÓL A HAL MÉRETÉIG

5.1. Szegmentálás és a két nézet párosítása

Az előtanított YOLOE-11m-seg a „fish” kategória alapján, mindkét rektifikált képen külön keresi a halakat [8]. Minden egyedhez bináris maszkot rendel, amely a hal felületéhez sorolt képpontokat jelöli. A maszkok a rektifikált képpel azonos méretűek és koordinátarendszerűek.

A párosítás a bal ($u; v$) pont jobb oldali helyét az $(u - d; v)$ koordinátán keresi. Előtte egy jobb-bal illesztés ellenőrzi, hogy a két irány ellentétes előjelű diszparitásának összege közel nulla-e. A kódban `valid` az ellenőrzött pontokat, `mask` és `right_mask` a vizsgált maszkpárt jelöli.

4. kódrészlet. Egyezési arány és megtartott képpontok (fish_pipeline.py; rövidítve)

```
selected = np.zeros_like(valid)
# ... a hiányzó maszkok ellenőrzése ...
yy, xx = np.nonzero(mask & valid & np.isfinite(disparity))
# ... az üres pontlista ellenőrzése ...
xr = np rint(xx - disparity[yy, xx]).astype(np.int64)
inside = (xr >= 0) & (xr < valid.shape[1])
accepted = np.zeros(len(xx), dtype=bool)
accepted[inside] = right_mask[yy[inside], xr[inside]]
selected[yy[accepted], xx[accepted]] = True
return float(np.count_nonzero(accepted) / len(xx)), selected
```

Az `xr` a jobb oldali vízszintes hely, az `inside` a képen belüli vetületeket választja ki. Az `accepted` jelzi a jobb maszkba eső találatokat. A visszaadott arány a párosítás pontszáma, a `selected` pedig a méretbecsléshez megtartott bal oldali pontkészlet. A program kölcsönösen legjobbat, a többi jelölttől kellően elkülönülő párt választ.

5.2. A pontfelhő hosszanti kiterjedése

A mélység szerinti kiugró pontok eltávolítása után a főkomponens-analízis (PCA) megkeresi a pontfelhő legnagyobb szórású irányát. A hosszbecslés e térbeli tengelyhez, így a hal helyzetéhez igazodik.

5. kódrészlet. A kijelölt felület PCA-alapú kiértékelése (measurement.py; rövidítve)

```

selected = (mask > 0) & result.valid_mask
points = result.xyz_mm[selected].reshape(-1, 3).astype(np.float64)
# ... a pontok ellenőrzése, szűrése és szükség esetén ritkítása ...
centroid = np.median(points, axis=0)
centered = points - centroid
covariance = np.cov(centered, rowvar=False)
eigenvalues, eigenvectors = np.linalg.eigh(covariance)
order = np.argsort(eigenvalues)[::-1]
axes = eigenvectors[:, order].T
projections = centered @ axes.T
low = np.percentile(projections, config.pca_low_percentile, axis=0)
high = np.percentile(projections, config.pca_high_percentile, axis=0)
dimensions = high - low

```

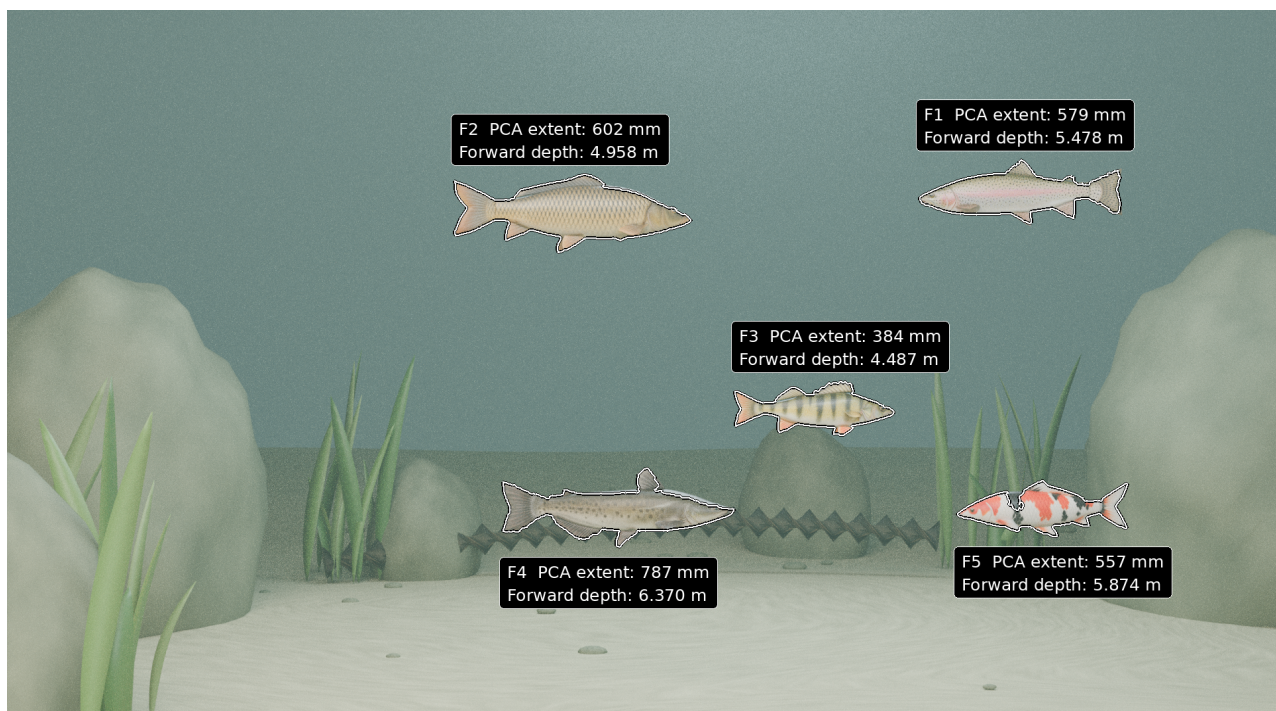
A mask a párosítás és a mélységszűrés után megmaradt kijelölés. A points az egy halhoz tartozó érvényes koordinátákat, a centroid azok koordinátáinkénti mediánját tárolja. A kovarianciamátrixból számított sajátvektorok (axes) adják a főirányokat, a projections pedig a pontok tengelyirányú vetületeit. A high-low alapbeállítás szerint az 1. és 99. percentilis közötti kiterjedés; ez mérsékli a szélső értékek hatását. A dimensions[0] lesz a jelentés length_mm értéke.

Az elfogadáshoz legalább 150 képpontos maszk, 100 térbeli pont, 30% érvényes pontarány, 60% jobb-maszk-támogatás és 80% hosszanti lefedettség kell. A párosítás pontszámkülönbsége legalább 0,15, a kétirányú diszparitás megengedett eltérése 1,5 képpont. Elutasításkor a program az okot rögzíti.

6. A GEOMETRIAI MŰKÖDÉS KÍSÉRLETI ELLENŐRZÉSE

6.1. Halakat tartalmazó Blender-jelenet

Az öt textúrázott halmodellt tartalmazó jelenetben a kamerabázis 1 m, a gyújtótávolság 50 mm, az érzékelőszélesség 36 mm, a képfelbontás 1920 × 1080 képpont. A párhuzamos kamerák 1,20 m magasan állnak. A 4.1. pont sakktáblás kalibrációja változatlan; a halas futás 640 képpontos diszparitástartományt és 25 000 mm legnagyobb mélységet használ. [10]



2. ábra. A halas jelenet bal nézete, a becsült PCA-hosszal és tengelyirányú mélységgel

Az F1–F5 jelölések az ábrát és a táblázatokat kapcsolják össze. A modellnevek a jelenetgenerátor adataiból származnak. A PCA-hossz a látható felület percentilisekkel határolt kiterjedése; a referenciahossz a nyugalmi modell teljes orr–farok kiterjedése. A becsült távolságok a felületi mediánhoz, a referenciák a modell referenciapontjához tartoznak. [11]

Referenciaértékek és sztereóbecslések

1. táblázat

Jel / modell	Lref [mm]	Lbecs [mm]	Yref [m]	Zbecs [m]	rL,ref [m]	rL,becs [m]
F1 pisztráng	620	578,7	5,5000	5,4776	5,6490	5,6263
F2 ponty	680	601,7	5,0000	4,9576	5,0224	4,9807
F3 sügér	420	384,1	4,5000	4,4868	4,5228	4,5105
F4 harcsa	850	787,5	6,4000	6,3698	6,4223	6,3929
F5 koi	600	557,2	5,9000	5,8735	6,0693	6,0426

6.2. A referenciaeltérések számítása

A referencia a generátorszkrét alapértelmezett 1. képkockája. A Blenderben X jobbra, Y előre, Z felfelé mutat; a bal kamera (0; 0; 1,20) m helyen áll. A távolság referenciája $rL,ref = \sqrt{(Xref^2 + Yref^2 + (Zref - 1,20)^2)}$. A becslés eltérése $\Delta q = q_{becs} - q_{ref}$, relatív értéke $\varepsilon q = 100 \cdot \Delta q / q_{ref}$. A negatív előjel alábecslést jelent. A táblázat Yref és Zbecs oszlopa az előre mutató mélységet jelöli a Blender, illetve az OpenCV koordináta-rendszerében.

A sztereóbecslések referenciaeltérései

2. táblázat

Jel	ΔL [mm]	εL [%]	ΔY [mm]	εY [%]	ΔrL [mm]	εrL [%]
F1	-41,26	-6,66	-22,37	-0,41	-22,71	-0,40
F2	-78,34	-11,52	-42,39	-0,85	-41,78	-0,83
F3	-35,93	-8,55	-13,17	-0,29	-12,35	-0,27
F4	-62,55	-7,36	-30,15	-0,47	-29,44	-0,46
F5	-42,79	-7,13	-26,48	-0,45	-26,76	-0,44

Az öt egyedi abszolút relatív eltérés átlaga a hossza 8,24%, a mélységre 0,49%, a bal kameratávolságra 0,48%. A hosszeltérések 6,66–11,52% közöttiek. A farok és az úszók hiányos rekonstruálása, a test hajlása és a szélső vetületek elhagyása egyaránt hozzájárulhat az alábecsléshez.

A $Z = fB/d$ összefüggés szerint kis diszparitásbizonytalanságra $\sigma Z \approx Z^2 \sigma d / (fB)$, ahol σd a diszparitás bizonytalansága. Kétszeres mélységnél így közel négyszeres a mélységi bizonytalanság. A próba gyújtótávolságával 0,5 képpontnyi változás az F3 mélységénél 3,77 mm, az F4-nél 7,61 mm mélységváltozást jelent.

7. KÖVETKEZTETÉSEK

A kalibrációból elkészül a kamerageometria és a Q mátrix, a képpont-párosításból pedig a diszparitás. Ezek együtt adják a térbeli koordinátákat. A párosított halmaszk kiválasztja az egyed pontjait, a PCA pedig meghatározza hosszanti kiterjedésüket.

A sztereó digitális képfeldolgozás felismeri a halak fajtáját és fejlettségi fokát. Ezzel a rendszerrel lehet működtetni egy intelligens és környezetbarát viselkedési sorompót a halak vízkivételi művekbe történő bejutásának megakadályozására.

KÖSZÖNETNYILVÁNÍTÁS

Dolgozatunkat a Romániai Oktatási és Kutatási Minisztérium (CCCDI - UEFISCDI) támogatta anyagiilag a PN-IV-P7-7.1-PTE-2024-0249 "LEADER" tudományos program/szerződés keretében.

IRODALMI HIVATKOZÁSOK

- [1] Gabriela Cîrciumaru, Rares-Andrei Chihaia, &al., *Experimental Study of a Fish Behavioral Barrier Based on Bubble Curtains for a River Water Intake, in Inland Waters*, Ecology, Limnology, and Environmental Protection, 03 January 2024. DOI: 10.5772/intechopen.1003997
- [2] Nicoleta-Oana Nicula, Eduard-Marius Lungulescu, *Barriers for Fish Guidance: A Systematic Review of Non-Physical and Physical Approaches*, Water 2026, 18(2), 225; <https://doi.org/10.3390/w18020225>
- [3] Noonan, M. J., Cushman, J. F., Noonan, B. P. , *Behavioral fish guidance barriers: A synthesis of effectiveness and future direction*, Fish and Fisheries, 2018. 19(4), 637-657
- [4] Gabriela Cîrciumaru, Rares, -Andrei Chihaia, &al. *Experimental Analysis of a Fish Guidance System for a River Water Intake*, Water 2022, 14, 370. <https://doi.org/10.3390/w14030370>
- [5] Cîrciumaru G., Dancă P. A., Chihaia R.-A., El-Leathay L.-A. PIV-Based Characterization of the Hydrodynamic Effects of a Bubble Curtain Intended for Fish Guidance. Water, 2026, 18(19), 2430. <https://doi.org/10.3390/w18192430>
- [6] OpenCV 4.13.0: Camera Calibration and 3D Reconstruction. https://docs.opencv.org/4.13.0/d9/d0c/group__calib3d.html (Elérés: 2026.10.05.)
- [7] OpenCV 4.13.0: cv::StereoSGBM Class Reference. https://docs.opencv.org/4.13.0/d2/d85/classcv_1_1StereoSGBM.html (Elérés: 2026.10.05.)
- [8] Ultralytics: YOLOE. Az átadott dokumentációban hivatkozott v8.3.221 modell ismertetése. <https://docs.ultralytics.com/models/yoloe/> (Elérés: 2026.10.05.)
- [9] Python <https://docs.python.org/3/>
- [10] Blender <https://docs.blender.org/>
- [11] Főkomponens-analízis (Principal Component Analysis, PCA) <https://hu.wikipedia.org/wiki/F%C5%91komponens-anal%C3%ADzis>