

Raport științific - Etapele I-II

Titlul proiectului: „*Inspectia scenelor subacvatice iluminate artificial folosind OpenROV*

Trident”

Codul proiectului: PN-III-P2-2.1-PED-2019-2805

Tip proiect: Proiect experimental - demonstrativ (PED)

Acronim: UW-ROV

Director proiect: Dr. Ing. Cosmin Ancuti

Universitatea Politehnica Timișoara, Facultatea de Electronică și
Telecomunicații, Departamentul MEO

E-mail: cosmin.ancuti@upt.ro

Website proiect: <http://ancuti.meo.etc.upt.ro/PED2019/>

Obiective proiect in Etapele I si II: *Implementare si optimizare metoda de imbunatatire a vizibilitatii in imagini subacvatice.*

Obiectivul principal al acestor etape este de a dezvolta o metoda eficienta de imbunatatire a vizibilitatii in imagini subacvatice. Asa cum a fost descris in propunerea de proiect am pornit de la prototipul publicat recent de directorul de proiect si echipa de cercetare [4]. Apoi, am investigat diverse strategii/metode de a optimiza aceasta metoda pentru a o putea integra pe platforma Open ROV Trident: Raspeberry Pi. Pentru atingerea obiectivului propus am parcurs etapele descrise in sectiunile urmatoare.

1 Implementare tehnica de imbunatatire a vizibilitatii in imagini subacvatice

1.1 Modelului optic in mediu subacvatic

Pentru un mediu de transmisie ideal, lumina este influențată în principal de proprietățile obiectelor țintă și caracteristicile lentilelor camerei. Acesta nu este cazul imaginilor subacvatice. În primul rând, cantitatea de lumină disponibilă în mediul subacvatic depinde de mai mulți factori. Interacțiunea dintre lumina soarelui și suprafața mării este afectată în principal de timpul zilei (care influențează unghiul de incidență a luminii) și de interfața dintre aer și apă. Ca urmare, obiecte subacvatice pot să fie percepute cu 25% mai mari decât sunt în realitate. Studiile complexe ale lui McGlamery [1] și Jaffe [2] au arătat că incidența totală de iradiere pe punctul generic al planului imaginii are trei componente principale în medii subacvatice: componentă directă, *forward scattering* și *backward scattering*.

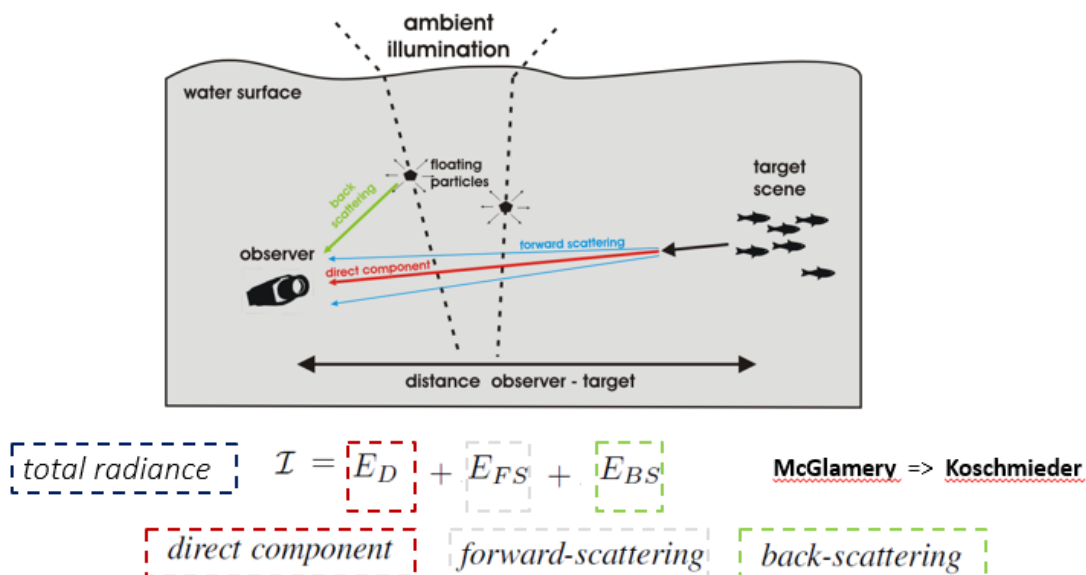


Figura 1 : Reprezentarea modelul optic subacvatic cu cele trei componente: componenta directă (direct component), componenta forward scattering și componeneta back scattering

Componenta directă este componenta de lumină reflectată direct de obiectul țintă pe planul imagine. Considerând coordonata x componenta directă este dată de expresia matematica:

$$E_D(x) = J(x)e^{-\eta d(x)} = J(x)t(x)$$

unde $J(x)$ este radianța obiectului, $d(x)$ este distanța între observator și obiect, și η este coeficientul de atenuare al mediului subacvatic.

Pe lângă absorbție, particulele plutitoare existente în mediile subacvatice determină, de asemenea, devierea (împrăștierea - *scattering*) razelor incidente ale luminii. Componenta *forward scattering* rezultă dintr-o deviație aleatorie a unei raze de lumină în drum spre obiectivul camerei. S-a determinat experimental că impactul componentei forward scattering poate fi aproximată de convoluția dintre componenta atenuată direct, cu o funcție de împrăștiere punct (*point spread function*) care depinde de distanța dintre planul imaginii și obiect. În altă ordine de idei, componenta *back scattering* se datorează luminii artificiale (de exemplu flash-ul camerei) care lovește particule de apă și se reflectă în camera digitala subacvatică. Componenta *back scattering* acționează ca un voal strălucitor suprapus peste imaginea obiectului țintă. Influența acestei componente poate fi redusă semnificativ prin simpla schimbare a poziției și unghiului luminii artificiale astfel încât cea mai mare parte a luminii de particule reflectate să nu ajungă la obiectivul camerei. Cu toate acestea, în multe cazuri practice, componenta back-scattering rămâne principala sursă de pierdere de contrast și de schimbare a culorii în imaginile subacvatice. Din punct de vedere matematic, este exprimată ca și:

$$E_{BS}(x) = B_{\infty}(x)(1 - e^{-\eta d(x)})$$

unde $B_{\infty}(x)$ este un vector de culoare cunoscut sub numele de componenta back-scattering.

Dacă se ignora componenta forward scattering, modelul optic subacvatic în general utilizat de metodele existente devine:

$$\mathcal{I}(x) = J(x)e^{-\eta d(x)} + B_{\infty}(x)(1 - e^{-\eta d(x)})$$

Acest model simplificat optic subacvatic are un aspect similar ca și formă matematică cu modelul lui Koschmieder [3], folosit pentru a caracteriza propagarea luminii în atmosferă.

Cu toate acestea, acest model nu reflectă cum coeficientul de atenuare depinde foarte mult pe lungimea de undă a luminii și, astfel, de culoarea din mediile subacvatice. Prin urmare, așa cum am demonstrat în lucrările publicate recent (rezultatele se prezintă și în figurile următoare)

metodele extinse folosind metodele de eliminare a ceții prezintă limitări serioase pentru imagini subacvatice înregistrate la mare adâncime, în prezența iluminării artificiale neuniforme și/sau absorbția selectivă a culorilor.

1.2 Tehnica de îmbunătățire a vizibilității în imagini subacvatice

În studiul nostru experimental, bazându-ne pe modelul optic descris în secțiunea anterioară, am pornit de la o tehnică [4] care a introdus recent o abordare nouă și eficientă, din punct de vedere al recuperării vizibilității din imaginile subacvatice, bazată pe o singură imagine capturată cu o cameră subacvatică convențională. Metoda se bazează pe fuziunea a două intrări derivate din imaginea inițială (o versiune procesată prin corectarea contrastului și cea de-a doua versiune echilibrată în alb -white balanced) care sunt combinate în final într-o singură imagine de ieșire. Procesul de fuziune folosește fuziunea multi-scală și este ghidat de trei măsuri pondere care permit filtrarea informației utile din cele două intrări derivate (vezi Figura 2).

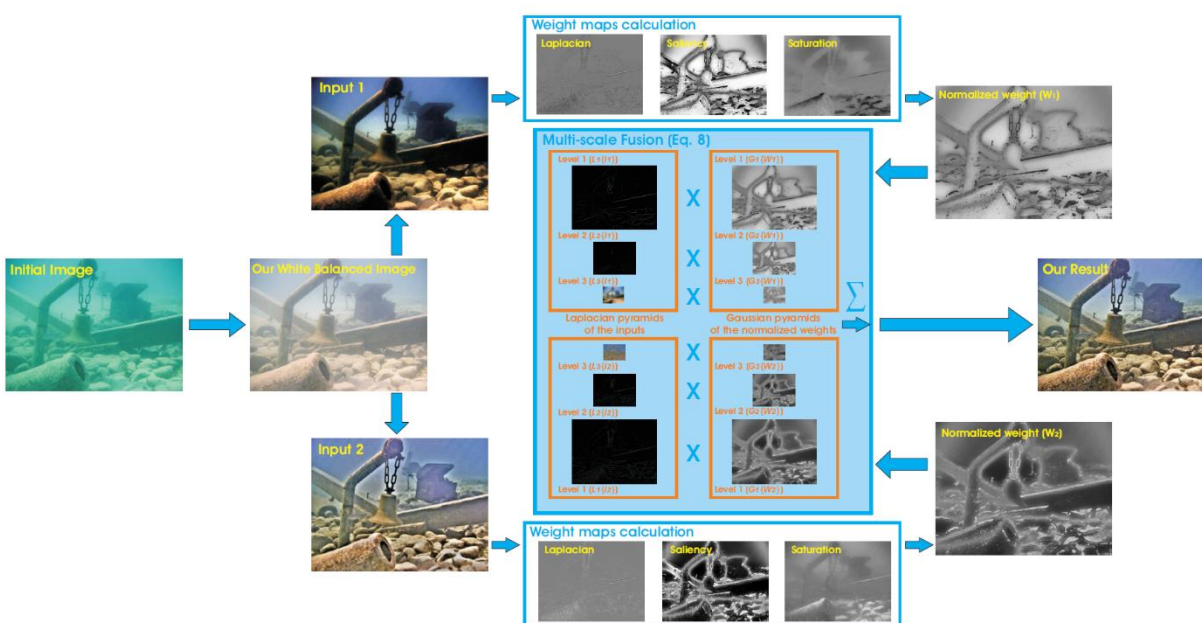


Figura 2. Imaginea de ansamblu a metodei de fuziune publicată în [4].

Una din marile probleme în imaginile subacvatice este atenuarea canalului de roșu. În studiul nostru pentru a compensa pierderea canalului roșu, ne-am bazat pe următoarele patru observații/principii:

- Canalul verde este relativ bine conservat in imaginile suacvatice, comparativ cu canalele de rosu și albastru. Lumina cu o lungime de undă lungă, adică lumina roșie, este cea care se atenuează mai întâi in mediul suacvatic;
- Canalul de verde este cel care conține informații despre culoarea oponenta în contrast cu canalul de roșu și, prin urmare, este deosebit de important să se compenseze atenuarea mai puternică indusă pe canalul de roșu, în comparație cu informația de pe canalul verde. Prin urmare, am compensat atenuarea pe canalul de roșu prin adăugarea unei fracțiuni din informația de pe canalul de verde la canalul de roșu. Am încercat inițial să adăugăm atât o fracțiune de verde, cât și de albastru la canalul de roșu, dar, folosirea doar a informațiilor din canalul de verde permite recuperarea mai bună a întregului spectru de culori, păstrând în același timp aspectul natural al fundalului in imaginile subacvatice;
- Compensarea ar trebui să fie proporțională cu diferența dintre valorile medii de verde și roșu, deoarece, în ipoteza „gray world” [5] (toate canalele au aceeași valoare medie înainte de atenuare), această diferență reflectă disparitatea / dezechilibrul dintre roșu și atenuarea canalului de verde;
- Pentru a evita saturarea canalului de roșu în timpul compensării acestuia, aceasta compensare ar trebui să afecteze în primul rând pixelii cu valori mici ale canalului de roșu și nu ar trebui să modifice pixelii care includ deja o componentă de roșu semnificativă. Din punct de vedere matematic, pentru a ține seama de observațiile de mai sus, canalul de roșu compensat I_{rc} la fiecare locație de pixel (x) este procesat după cum urmează:

$$I_{rc}(x) = I_r(x) + \alpha (\bar{I}_g - \bar{I}_r) (1 - I_r(x)) I_g(x)$$

unde I_r , I_g reprezintă canalele de culoare roșu și verde ale imaginii I , fiecare canal fiind în intervalul $[0, 1]$, după normalizarea prin limita superioară a intervalului lor dinamic și α reprezintă un parametru constant.

Pentru a obține prima intrare derivată, s-a efectuat o corecție gamma a versiunii de imagine echilibrată de alb (white-balancing). Corecția Gamma vizează corectarea contrastului global și este relevantă deoarece, în general, imaginile subacvatice cu echilibru de alb tind să pară prea luminoase. Această corecție crește diferența dintre regiunile mai întunecate/mai deschise cu

prețul unei pierderi de detalii în regiunile sub/supraexpuse. Pentru a compensa această pierdere, am calculat o a doua intrare care corespunde unei versiuni clare a imaginii echilibrate de alb. Prin urmare, folosind principiul „unsharp mask” am combinat o versiune neclară (în acest caz, filtrată gaussian) a imaginii cu imaginea rectificată.

Hartile ponderi (*weight maps*) sunt utilizate în timpul procesului de fuziune astfel încât pixelii cu o importanță mai mare să fie mai reprezentați în imaginea finală. Aceste ponderi sunt definite pe baza unui număr de valori locale. În tehnica analizată s-au folosit trei ponderi: contrastul laplacian (W_L), ponderea de proeminentă (W_S) și ponderea de saturație (W_{Sat}).

Deoarece implementarea naivă a fuziunii (combinarea directă a intrărilor și a hărților ponderi) introduce artefacte de halo neplăcute, am folosit în schimb descompunerea laplaciană multi-scală. Folosind același număr de niveluri, piramidele Gaussian și Laplacian sunt fuzionate independent la fiecare nivel:

$$\mathcal{R}_l(x) = \sum_k G_l \{ \bar{W}^k(x) \} L_l \{ \mathcal{I}_k(x) \}$$

unde l este nivelul piramidei, k este indicele de intrare, $L \{I\}$ denotă laplacianul intrării I și $G \{W\}$ este harta de pondere normalizată gaussian. Aici, normalizarea asigură că suma ponderilor pe cele trei intrări este egală cu unitatea pentru fiecare pixel. Rezultatul final fuzionat R este procesat prin însumarea contribuțiilor de la toate nivelurile calculate ale piramidei.

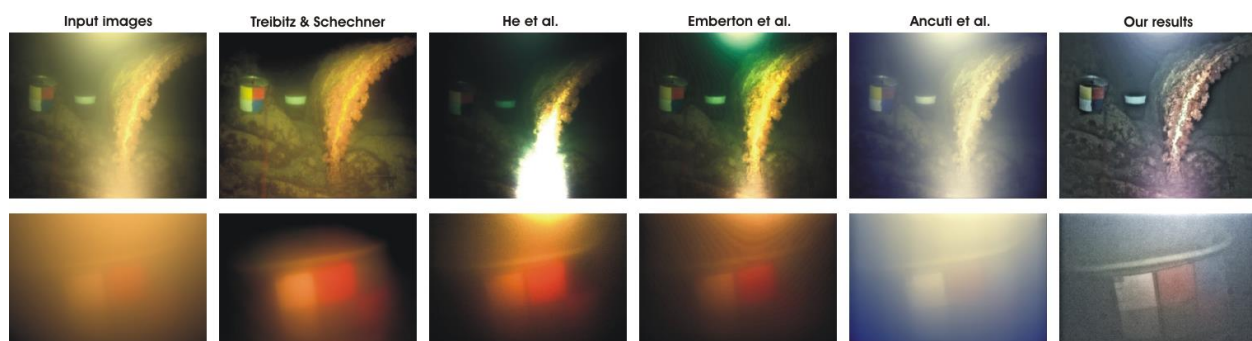


Figura 3: Rezultate comparative obținute pentru două imagini subacvatice. Ne-am comparat cu metodele [6],[7],[8] și [9].

2. Optimizare metoda de imbunatatire a vizibilitatii in imagini subacvatice

Una dintre principalele probleme pentru algoritmi existenți care utilizează fuziunea pe mai multe nivele este legată de abilitatea de a o utiliza pentru aplicațiile în timp real. În practică, pentru o fuziune cu imagini de înaltă rezoluție sunt necesare atât memorie substanțială cât și putere de calcul. Motivul principal este creșterea numărului de nivele corelată cu dimensiunea imaginilor de intrare din algoritmul de fuziune, deoarece algoritmul de fuziune multi-nivel necesită o compunere a piramidei Laplacian pentru imaginile de intrare și piramidă Gaussian pentru hărțile de ghidare a procesului de fuziune.

Pentru eficientizarea procesului, s-au dezvoltat procesoare de tip *fusion-exclusive*, dar hardware-ul exclusiv de fuziune introduce restricții cum ar fi lipsa extensibilității sistemului datorită deficienței memoriei, creșterea prețurilor și imposibilitatea modificării algoritmului.

De aceea, pentru a putea integra algoritmul pe un Raspberry Pi am cautat alte strategii care să evite procesul de fuziune multi-scala. Pentru a estima transmisia din modelul optic am aplicat DCP (Dark Channel Prior) [7]. Deși introdus pentru problema de *dehazing* (eliminarea cetei), această estimare generează rezultate satisfacatoare și pentru scene subacvatice în care atenuarea selectivă nu afectează semnificativ canalele de culoare. Pe baza DCP, transmisia $t(x)$ este calculată prin găsirea celei mai mici valori dintr-o vecinătate a unei locații x :

$$t(x) = 1 - \min_{y \in \Omega(x)} \left(\min_{c \in r, g, b} \mathcal{I}^c(x) / B_{\infty}^c \right)$$

unde $\Omega(x)$ este vecinătatea locației x din imagine iar B_{∞} reprezintă constanta de *backscattering*. Analizând diverse scene subacvatice, am observat că aceasta necunoscută a modelului optic poate fi aproximată în mod rezonabil când valoarea B_{∞} este înlocuită cu un vector de intensitate a culorii cu valoare maximă $[1; 1; 1]$:

$$t(x) \approx 1 - \min_{y \in \Omega(x)} \left(\min_{c \in r, g, b} \mathcal{I}^c(x) \right)$$

Pe dealta parte valoare de *backscattering* B_{∞} (cealalta necunoscuta a modelului optic) reprezintă lumina care este împrăștiata înapoi de particulele plutitoare din mediul subacvatic. Aceasta componenta este considerata a fi principala cauza care genereaza pierderea contrastului și degradarea culorii. Pentru scene subacvatice nu foarte distorsionate se poate calcula o valoare globală constantă pe intraga imagine. Inspirat din tehnica [10] am derivat urmatoarea valoare globala pentru calculul *backscattering*:

$$B_{G_{\infty}}^c = \max_{y \in M_{DC}^I} \left(\min_{z \in \Omega(y)} \mathcal{I}^c(z) \right)$$

unde M_{DC}^I reprezinta un set de locatii unde valoare de dark channel atinge maximul.

Pentru scene subacvatice mai complexe care au vizibilitatea foarte distorsionata putem calcula o valoare locala de backscattering:

$$B_{L_{\infty}}^c(x) = \max_{y \in M_{DC}^{\Psi(x)}} \left(\min_{z \in \Omega(y)} \mathcal{I}^c(z) \right)$$

unde $\Psi(x)$ reprezinta o regiune a imaginii centrata pe locatia x. Dimensiunea regiunii Ψ este în general mai mare decât dimensiunea regiunii Ω . În lucrarea noastra [10] am demonstrat experimental ca valoarea implicită a raportului dintre dimensiunea celor doua regiuni este 2. Cu toate acestea, pentru scene subacvatice de mică adâncime unde iluminarea este destul de uniformă, valoarea raportului poate fi setată la valori mai mari. Stabilirea unei dimensiuni optime a regiunii Ψ este totusi destul de complexa . O dimensiune mai mare este utila pentru scene cu mai multe surse de lumină în timp ce o dimensiune mai mica ar putea reduce eficienta de eliminare a distorsiunii de ceață (care afecteaza considerabil imaginile subacvatice).

Pentru a calcula o dimensiune optimă a regiunii, se pot estima două valori complementare ale valorii locale de *backscattering* (o estimare pentru o dimensiune mare a regiunii și o estimare pentru a dimensiune mică a acesteia). In acest caz valoarea optimă de backscattering local este calculata prin medierea acestor două valori.

3. Integrare metoda de imbunatatire a vizibilitatii in imagini subacvatice pe platforma hardware OpenROV

Varianta care am implementat-o si integrat-o pe platforma hardware a Open ROV Trident (Raspberry Pi) se bazeaza pe Darck Channel Prior (descrisa in sectiunea anterioara) si calculeaza o valoare unica a backscattering dar contine si un pas de preprocesare care utilizeaza operatorul 3C [11], foarte util in procesarea imaginilor subacvatice.

Modelul actual de **Open ROV Trident** este bazat pe **Raspberry Pi** care are un procesor quad-core si are un excelent suport periferic. În comparație cu Beaglebone Black (care a fost folosit pentru toate OpenROV-urile de până acum) care are un procesor AM3358 de 1GHz și o memorie DDR3 RAM de 512MB la bord, **Raspberry Pi** este un pas semnificativ în performanță și capacitate.

Open ROV Trident are următoarele caracteristici:

Dimensiuni: [L X W X H] 410 mm [16.14 in] X 205 mm [8.07 in] X 86 mm [3.37 in]

Greutate: 3.40 kg

Adâncime maximă: 100 m [328 ft]

Viteza maximă: 2 m/s [3.89 kts / 7.2 kph / 4.47 mph]

Baterie: 3S4P Li-NMC 18650 cells with built-in PCM, UN 38.3 , capacitate 95 Wh;
timp de incarcare 1.5 hr from 20% to 80%, 3 hr for 0% to 100%

Timp de rulare: 3 - 4 ore in condiții normale;

Putere consumata: 30 W nominal (10 W min, 180 W max)

Comunicare și interfața: WiFi 802.11 b/g/n; Android folosind OpenROV App.

Temperatura apei recomandată: -2°C to 40°C

Barometru intern

Raspberry Pi face parte din familia minicalculatoarelor (platforme cu microprocesor) care a fost dezvoltată inițial de către Raspberry Foundation pentru promovarea tehnologiei calculatoarelor în școlile din țările în curs de dezvoltare. Acesta deși are dimensiunea unui card de credit are caracteristicile unui PC fiind posibil conectarea acestuia la monitor și la tastatură.

Modelele Raspberry Pi sunt bazate pe procesoare ARM dezvoltate de Broadcom. Modelele Broadcom aflate în componenta diverselor modele Raspberry PI sunt procesoare multimedia de

tipul system-on-chip (SoC) care au incluse pe un singur chip atât procesorul central (CPU) dar și procesorul grafic (GPU) și celelalte componente hardware auxiliare. Acestea au o arhitectură ARM care este folosită cu preponderență pentru telefoanele smart (smartphone). Procesoarele ARM necesită un consum de energie electrică redus și au un set de instrucțiuni simplificat (RISC) comparativ cu procesoarele din familia x86. Aceste caracteristici permit o funcționare normală folosind o sursă de energie de doar 5V și 1 A care în cazul Raspberry Pi se conectează printr-un port (onboard) micro-USB.

Spre deosebire de PC-urile uzuale, sistemul de calcul Raspberry Pi folosește un sistem de operare special. Dacă majoritatea PC-urilor (peste 95%) utilizează sistemele de operare Windows de la Microsoft sau OS X de la Apple, Raspberry Pi se bazează în principal pe sistemul de operare Raspbian, o versiune a sistemului de operare Linux (Debian). Totuși Raspberry Pi permite instalarea și a altor sisteme de operare.

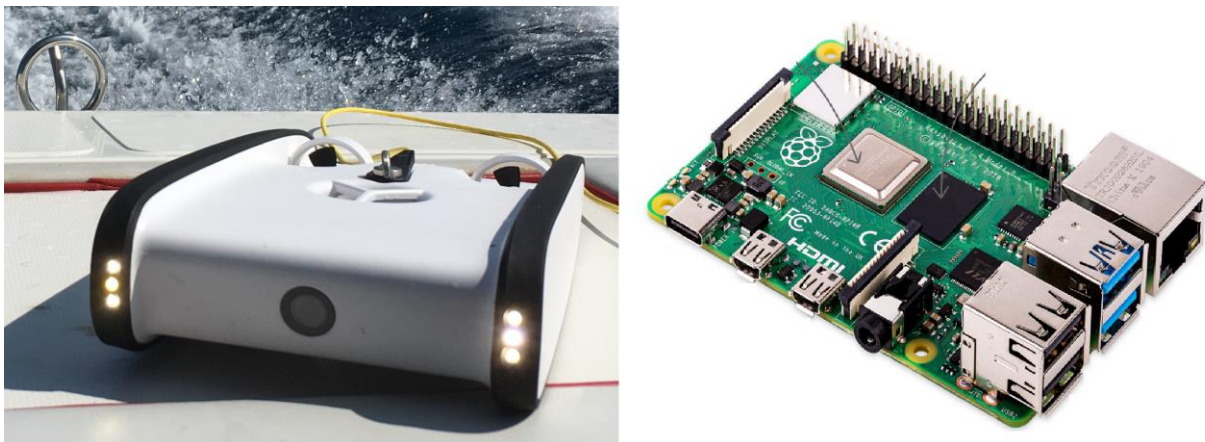


Figura 4: Open ROV Trident si Raspberry Pi 4

In acest studiu experimental am integrat aplicatia pe versiunea Raspberry Pi 4. Raspberry Pi 4 acceptă procesor ARM Cortex-A72 quad-core de 1,5 GHz pe 64 de biți (ARM v8, BCM2837), până la 8 GB RAM LPDDR2, încorporat WiFi și suport excelent pentru periferice.

Pentru implementarea metodei de îmbunătățire a vizibilității am instalat **biblioteca C++ OpenCV** pe platforma Raspberry Pi. **OpenCV** este o bibliotecă extrem de folosită în comunitatea de computer vision fiind dezvoltată de compania Intel. S-a dovedit a fi foarte utilă în aplicațiile de prelucrarea a imaginilor în timp real. Este folosită foarte mult pe calculatoare normale în diverse aplicații computer vision și de prelucrare a imaginilor și secvențelor de imagini. Open CV include funcții și implementări optimizate ale unor metode recent publicate dar prezintă și un concept nou în interfața de programare. Limbajul principal folosit este C++, dar există și versiuni de legătură pentru Python, Java sau MatLab. Toate dezvoltările curente și algoritmi se fac pentru crearea noilor interfețe cu sistemele de operare și pentru minimizarea resurselor folosite în timp real.

Open CV rulează pe diferite platforme printre care și Windows, Linux, Android, etc. În procesul de compilare OpenCV folosește Cmake, o platformă care este concepută pentru realizarea cadrului aplicațiilor care depind de mai multe biblioteci. Platforma are dependențe minime care necesită doar un compilator C++ în propriul sistem de dezvoltare.

Instalarea librăriei Open CV in C++ pe Raspberry Pi : În acest scop trebuie utilizată opțiunea de utilizare a spațiului total de pe micro-SD card prin opțiunea Expand File System. Este necesar ca primul pas să se instaleze CMake-ul pentru configurarea procesului de compilare OpenCV. Se pot instala câteva biblioteci adiționale pentru suportul diferitelor formate JPEG, PNG, TIF, etc (libjpeg, libtiff5, libpng12), pentru lucrul cu fișierele video (libavcodec, libavformat, libswscale, libxvidcore) precum și câteva dependențe pentru optimizarea lucrului cu matrici. După ce aceste dependențe sunt instalate se face download la cod și apoi se compilează cu Cmake.

4. Testare metoda de imbunatatire a vizibilitatii in imagini subacvatice

Metoda de imbunatatirea vizibilitatii in imagini subacvatice a fost testată pentru zeci de imagini subacvatice diferite (a se vedea cateva rezultate comparative in Figurile 5 si 6). Pentru a valida abordarea noastră, am comparat rezultatele obtinute cu câteva metode specializate incluzand tehnicile lui Galdran et al [13], Drews-Jr et al. [12], Emberton et al. [9] și Berman et al [14].

În Figurile 5 si 6 sunt prezentate câteva imagini subacvatice. Pentru evaluarea imaginilor subacvatice nu exista la ora actuala o metrica general acceptata de comunitate si prin urmare ne-am rezumat doar la o comparare calitativa a imaginilor rezultate. Printr-o inspecție atentă, se poate observa ca metoda noastră genereaza rezultate comparative și chiar mai bune comparativ cu tehnicile evaluate.

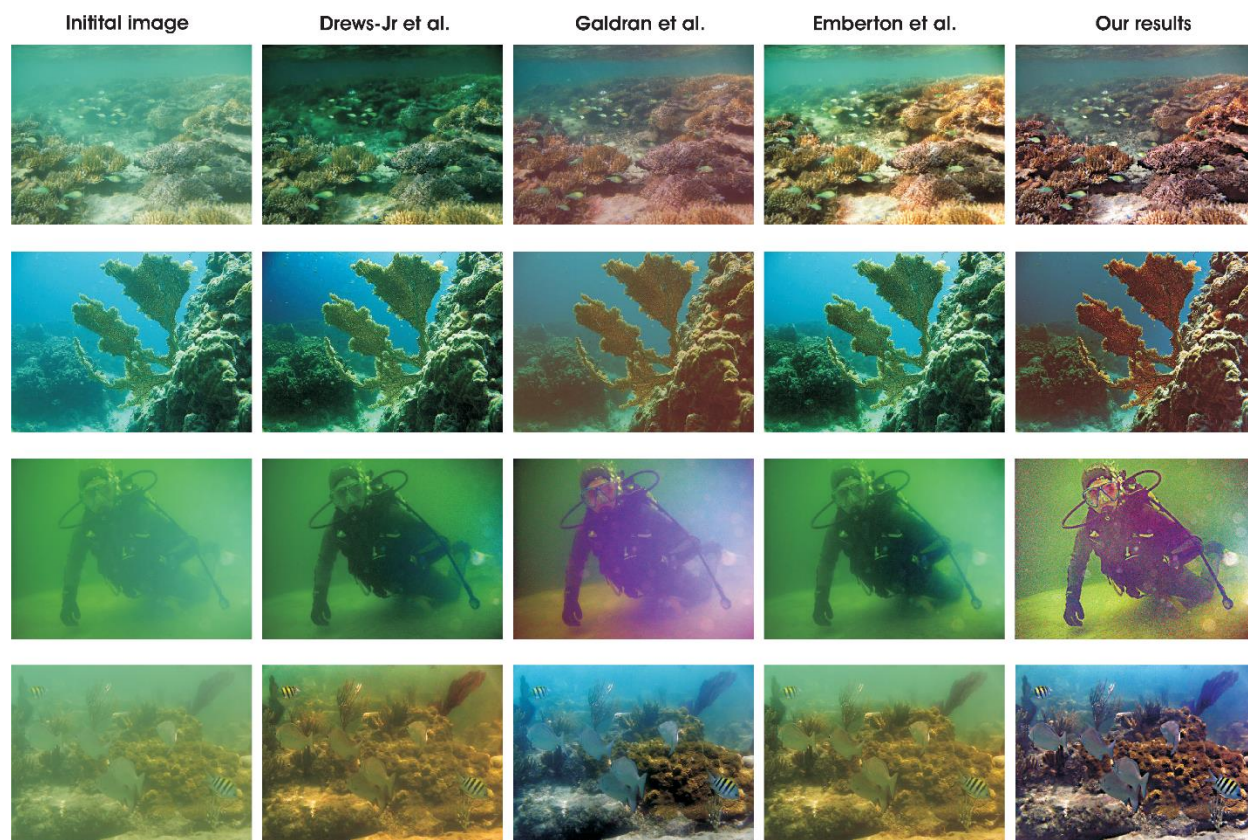


Figura 5: Rezultate comparative obtinute cu metoda noastra publicata recent [11]. Ne-am comparat cu metodele [12],[13] si [9].

Unul dintre aspectele importante, așa cum am prezentat anterior, este că metoda subacvatică prezentată anterior a fost implementată pe Raspberry Pi 4 folosind biblioteca OpenCV. În experimentele noastre am testat imagini o rezoluție spațială de 640x480. În studiul nostru experimental, metoda implementată de îmbunătățire a vizibilității în imagini subacvatice procesează imagini cu o rezoluție spațială de 640x480 în aproximativ 20-30 ms. De asemenea, am observat că atunci când creștem rezoluția spațială a imaginilor, timpul de procesare folosind

implementarea noastra creste relativ liniar (e.g. 1200x800 am obtinut timpi de procesare de aproximativ 3-4 ori mai mari decat la rezolutia de 640x480)

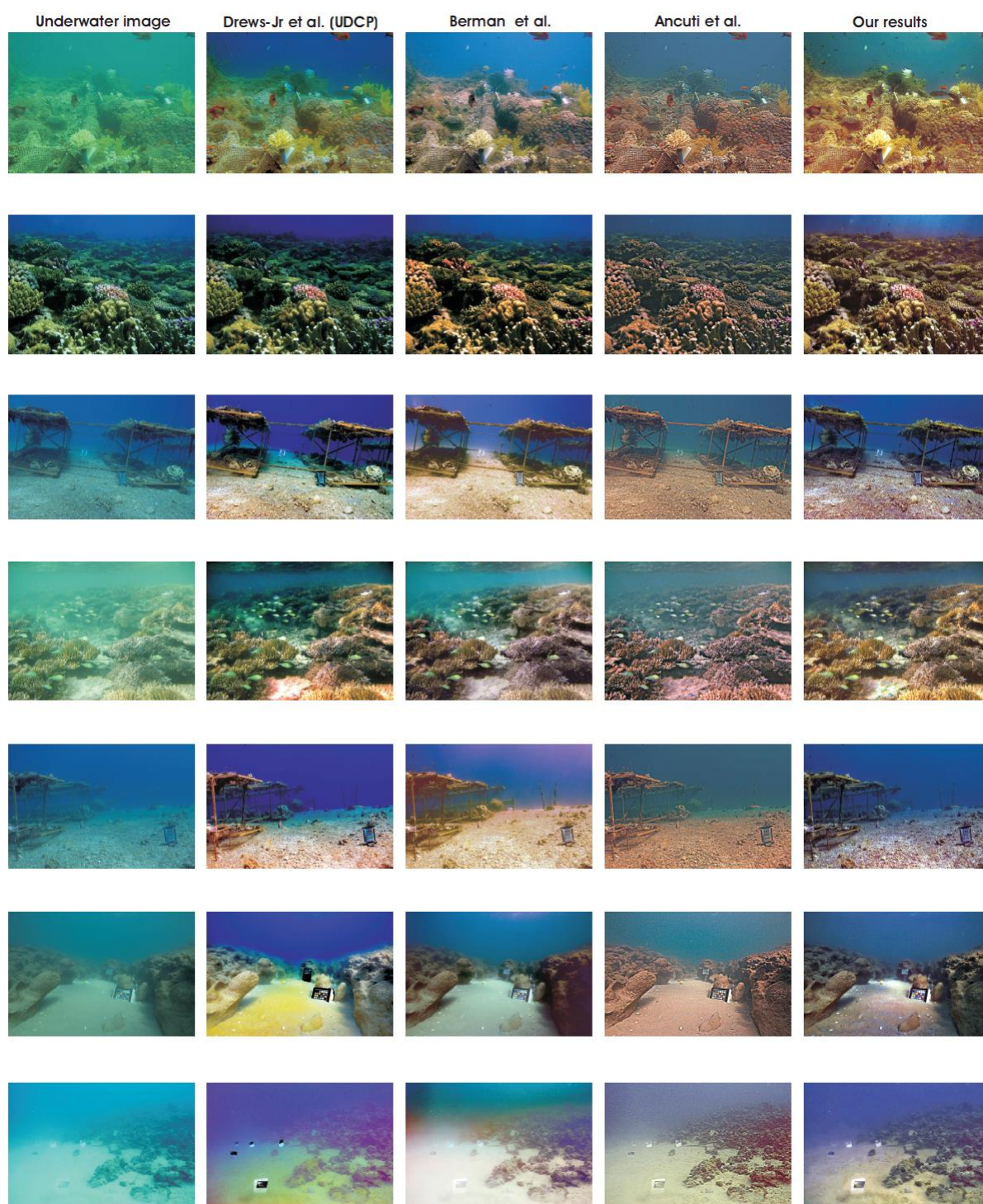


Figura 6. Rezultate comparative obtinute cu metoda noastra publicata recent [11]. Ne-am comparat cu metodele [12],[14] si [4].

5. Participare la conferințe în Etapa I și II a proiectului:

În cea de-a doua etapă am participat (online) la 2 conferințe internaționale:

1. **IEEE ELMAR 2021, 13-15 septembrie, Zadar Croatia.**
2. **IEEE OCEANS 2021, 23-26 septembrie, San Diego, US.**

6. Rezultate Etapa I+II

În această etapă rezultatele au fost publicate 3 lucrări ISI/WoS:

- Arpad Kis, Horia Balta and Cosmin Ancuti, „*Underwater image enhancement on low-cost hardware platform*”, IEEE ELMAR, Zadar, Croatia, septembrie 2021
- Horia Balta and Codruta O. Ancuti and Cosmin Ancuti, „*Effective underwater image restoration based on color channel compensation*”, IEEE Oceans, San Diego, US, septembrie 2021
- M. Sbert, Cosmin Ancuti, Codruta O. Ancuti, J. Poch, S. Chen, M. Vila, „*Histogram Ordering*”, IEEE Access 2021, revista Q1 ISI/WOS

Referințe bibliografice:

- [1] B. L. McGlamery, “A computer model for underwater camera systems,” in *Proc. SPIE 0208, Ocean Optics VI*, October 1979.
- [2] J. S. Jaffe, “Computer modeling and the design of optimal underwater imaging systems,” *IEEE Journal of Oceanic Engineering*, vol. 15, pp. 101 – 111, April 1990.
- [3] H. Koschmieder, “Theorie der horizontalen sichtweite,” *Beiträge zur Physik der freien Atmosphäre*, 1924.
- [4] C. O. Ancuti, C. Ancuti, C. De Vleeschouwer, and P. Bekaert, “Color balance and fusion for underwater image enhancement,” in *IEEE Transactions on Image Processing*, 2018.

- [5] G. Buchsbaum, "A spatial processor model for object colour perception," J. Franklin Inst., vol. 310, no. 1, pp. 1–26, Jul. 1980.
- [6] T. Treibitz and Y.Y. Schechner, "Active polarization descattering," IEEE Trans Pattern Anal Mach Intell., 2009.
- [7] K. He, J. Sun and X. Tang, Single image haze removal using dark channel prior, IEEE TPAMI, 2011
- [8] S. Emberton, L. Chittka, and A. Cavallaro, "Hierarchical rank-based veiling light estimation for underwater dehazing," BMVC, 2015.
- [9] C. Ancuti, C. O. Ancuti, T. Haber, and P. Bekaert, "Enhancing underwater images and videos by fusion," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012.
- [10] C. Ancuti, C. O. Ancuti, C. De Vleeschouwer, R. Garcia, and A. C. Bovik, "Multi-scale underwater descattering," Int. Conf. on Pattern Recognition (ICPR), 2016.
- [11] Horia Balta and Codruta O. Ancuti and Cosmin Ancuti, „Effective underwater image restoration based on color channel compensation”, IEEE Oceans, San Diego, US, septembrie 2021
- [12] P.L.J. Drews Jr., E. R. Nascimento, S.S.C. Botelho, and M. F. M. Campos, "Underwater depth estimation and image restoration based on single images," IEEE CGA, 2016.
- [13] A. Galdran, D. Pardo, A. Picon, and A. Alvarez-Gila, "Automatic redchannel underwater image restoration," Journal of Visual Communication and Image Representation, 2015.
- [14] D. Berman, T. Treibitz and S. Avidan, "Diving into haze-lines: Color restoration of underwater images," Proc. of British Machine Vision Conference (BMVC), 2017