

Concepțe și Aplicații în Vederea Artificială - Tema 2

Detectarea și recunoașterea facială a personajelor din serialul de desene animate Scooby-Doo

Obiectiv

Scopul acestei teme este implementarea unui sistem automat de detectare și recunoaștere facială a personajelor din serialul de desene animate *Scooby-Doo* folosind algoritmi de Vedere Artificială discutați la curs și implementați parțial la laborator.

Scooby-Doo

Scooby-Doo, Where Are You! este un serial animat creat de Joe Ruby și Ken Spears și produs de Hanna-Barbera, care a debutat în 1969. Serialul urmărește aventurile unui grup de adolescenți detectivi, cunoscuți sub numele de **Mystery Inc.**, care călătoresc împreună pentru a rezolva mistere aparent supranaturale. Echipa este formată din **Fred**, liderul grupului, curjos și organizat, **Daphne**, elegantă și curioasă, **Velma**, inteligentă și analitică, și cei mai neobișnuiti membri ai echipei: **Shaggy** și câinele său, **Scooby-Doo**. Aceștia doi sunt extrem de fricoși, dar au un talent special pentru a ajunge, fără să vrea, exact în mijlocul pericolului. Misterele ii poartă pe eroi prin conace bântuite, parcuri de distractii abandonate, castele sau muzee, unde întâlnesc monștri, fantome și creațuri înfricosătoare. Deși totul pare supranatural la început, fiecare caz se dovedește a avea o explicație logică, iar „monstrul” este de fapt un om mascat, cu un plan ascuns. Serialul combină suspansul cu umorul, mai ales prin reacțiile exagerate ale lui Scooby-Doo și Shaggy, și transmite mesajul că inteligența, munca în echipă și curajul pot învinge orice mister.

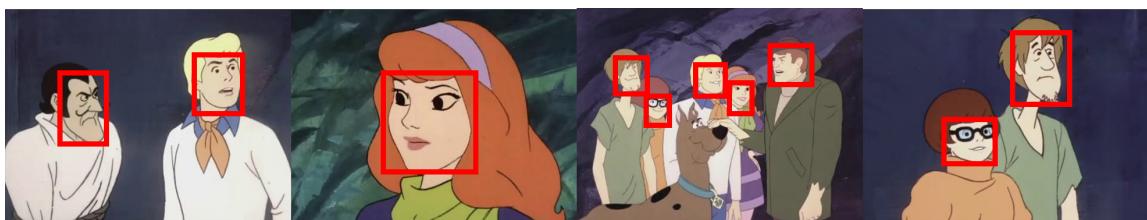


Figura 1: Detectare facială a personajelor din *Scooby-Doo*: fiecare față de interes este adnotată cu o ferestru dreptunghiulară de culoare roșie.



Figura 2: Recunoașterea facială a personajelor din *Scooby-Doo*: fiecare față de interes este adnotată cu o fereastră dreptunghiulară de culoare specifică clasei personajului (albastru - Fred, galben - Daphne, verde - Shaggy, violet - Velma).

Task 1 - detectarea facială

Prima problemă pe care o aveți de rezolvat constă în detectarea facială a tuturor fețelor personajelor care apar în imagini. Pentru fiecare imagine de intrare algoritmul vostru trebuie să returneze o mulțime de detecții asociate (fereastră dreptunghiulară și scor) ce localizează *toate* fețele dintr-o imagine. Figura 1 arată câteva exemple din mulțimea de antrenare și adnotările corespunzătoare, ce constau în ferestre dreptunghiulare de culoare roșie ce încadrează perfect *fiecare față*.

Task 2 - recunoaștere facială

A doua problemă pe care o aveți de rezolvat constă în recunoașterea facială a **anumitor personaje**. În serialul *Scooby-Doo*, alături de Scooby-Doo, apar cu preponderență alte patru personaje: **Fred**, **Daphne**, **Shaggy** și **Velma**. Vom considera problema recunoașterii faciale numai pentru aceste patru personaje. Pentru fiecare imagine de intrare, algoritmul vostru trebuie să returneze o mulțime de detecții asociate, fiecare constând în:

- numele personajului detectat (Fred, Daphne, Shaggy sau Velma),
- fereastra dreptunghiulară care încadrează față,
- scorul detecției.

Figura 2 arată câteva exemple din mulțimea de antrenare și adnotările corespunzătoare, ce constau în ferestre dreptunghiulare ce încadrează perfect *fețele de interes*. Fiecare detecție are o culoare specifică clasei personajului (albastru - Fred, galben - Daphne, verde - Shaggy, violet - Velma).

Descrierea datelor

Arhiva cu materiale (disponibilă aici: <https://tinyurl.com/CAVA-2025-TEMA2>) conține patru directoare: *antrenare*, *validare*, *testare* și *evaluare*.

Directorul *antrenare* conține datele de antrenare. Pentru fiecare din cele patru personaje (Fred, Daphne, Shaggy, Velma) există un director corespunzător cu 1000 imagini de antrenare. Fiecare imagine conține adnotate toate fețele personajelor care apar în imagine.

Content of fred_annotations.txt:

```

1 |0001.jpg 158 128 215 211 daphne
2 |0001.jpg 276 58 353 154 fred
3 |0002.jpg 122 59 183 143 fred
4 |0002.jpg 290 152 354 217 velma
5 |0003.jpg 193 103 258 192 fred
6 |0004.jpg 209 89 303 234 fred
7 |0005.jpg 234 66 281 131 fred
8 |0006.jpg 278 77 353 193 fred
9 |0007.jpg 55 215 133 294 velma
10 |0007.jpg 183 118 253 224 fred
11 |0007.jpg 301 185 368 266 daphne
12 |0008.jpg 180 36 221 100 fred
13 |0009.jpg 75 82 99 123 fred
14 |0009.jpg 148 83 175 112 daphne
15 |0009.jpg 226 119 255 150 velma
16 |0009.jpg 340 109 365 150 unknown
17 |0009.jpg 261 159 277 197 unknown
18 |0010.jpg 106 112 142 159 unknown
19 |0010.jpg 186 162 227 209 velma
20 |0010.jpg 291 89 328 143 fred
21 |0011.jpg 98 84 189 228 shaggy
22 |0011.jpg 289 154 379 290 fred

```

Content of daphne_annotations.txt:

```

1 |0001.jpg 155 135 215 210 daphne
2 |0001.jpg 285 46 344 134 fred
3 |0002.jpg 40 134 86 184 daphne
4 |0002.jpg 193 108 236 169 fred
5 |0002.jpg 378 132 421 198 shaggy
6 |0002.jpg 95 229 148 283 velma
7 |0003.jpg 97 178 212 285 velma
8 |0003.jpg 262 156 368 257 daphne
9 |0004.jpg 101 55 121 84 shaggy
10 |0004.jpg 189 99 209 126 fred
11 |0004.jpg 243 129 264 154 velma
12 |0004.jpg 348 70 372 100 daphne
13 |0005.jpg 144 100 214 210 unknown
14 |0005.jpg 320 113 367 199 fred
15 |0005.jpg 417 156 455 217 daphne
16 |0006.jpg 53 136 89 180 velma
17 |0006.jpg 169 93 212 158 daphne
18 |0007.jpg 122 129 167 179 velma
19 |0007.jpg 250 110 291 165 daphne
20 |0007.jpg 354 81 389 144 fred
21 |0008.jpg 109 86 159 170 shaggy
22 |0008.jpg 207 164 266 217 velma

```

Figura 3: Fiecare feță adnotată din imaginile de antrenare îi corespunde o linie în fișierul text corespunzător cu adnotări ce urmează formatul nume_imagine xmin ymin xmax ymax nume_personaj.

Fiecare feță adnotată îi corespunde o linie din fișierul text corespunzător (Figura 3) și are formatul:

nume_imagine xmin ymin xmax ymax nume_personaj

unde:

- nume_imagine reprezintă numele imaginii din directorul corespunzător personajului;
- xmin ymin xmax ymax reprezintă coordonatele ferestrei dreptunghiulare ce încadrează fața în imagine. Colțul din stânga sus are coordonatele (xmin, ymin) iar colțul din dreapta jos are coordonatele (xmax, ymax).
- nume_personaj reprezintă numele personajului ce are față adnotată. Folosim pentru adnotarea fețelor cinci clase posibile: *fred*, *daphne*, *shaggy*, *velma* și *unknown* (pentru toate fețele personajelor diferite de cele patru de interes).

În total sunt 4000 imagini de antrenare ce conțin 6547 fețe adnotate.

Directorul *validare* conține datele de validare. Acestea constau din 200 de imagini adnotate în formatul descris mai sus. Puteți să vă folosiți de aceste date de validare pentru a estima performanțele diverselor voastre soluții.

Directorul *testare* are aceeași structură ca directorul *validare*. Vom face publice datele de test după **prima fază** (detalii mai târziu). Vom evalua performanța algoritmului vostru pe 200 de imagini de test, similară ca distribuție a fețelor cu imaginile de validare.

Directorul *evaluare* vă indică cum să vă scrieți codul astfel încât să respectați formatul fișierelor cu rezultate impus pentru faza de evaluare pe datele de test (**faza a doua**) ce va avea loc după trimiterea codului cu soluția fiecărui student. Conține următoarele sub-directoare:

- *fake_test* - acest director exemplifică cum vor arăta datele de testare, el păstrează aceeași structură ca cea descrisă pentru directorul *validare* descris anterior. Acest director va fi similar cu directorul *testare* în care vom pune imaginile de testare pentru faza a două de evaluare.
- *fisiere_solutie* - acest director exemplifică formatul fișierelor cu rezultatele pe care trebuie să le trimiteți în faza a două. Veți trimite rezultatele voastre în acest format, încărcând o arhivă zip a unui director similar cu cel numit *331_Alexe_Bogdan*;
- *cod_evaluare* - acest director conține codul care va fi folosit pentru evaluarea automată a rezultatelor voastre folosind adnotările soluțiilor corecte (ferestrele ground-truth). Asigurați-vă că acest cod rulează pe fișierele voastre. Puteți folosi în acest sens datele de validare. Adnotările soluțiilor corecte vor fi disponibile după faza a două.

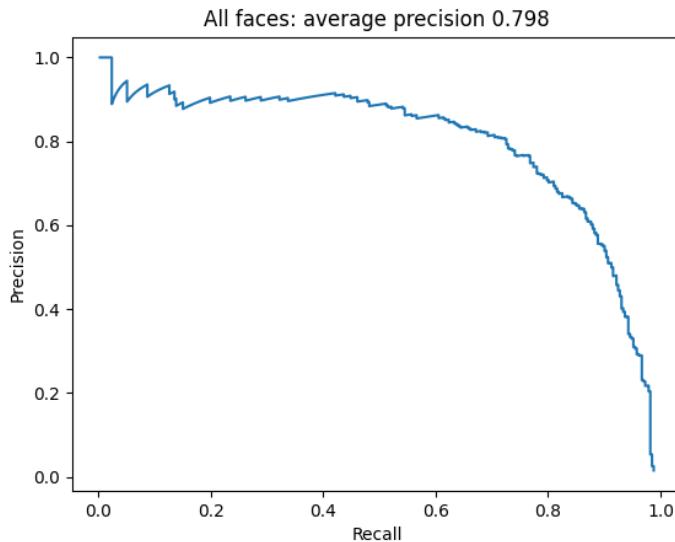


Figura 4: **Grafic precizie-recall pentru task-ul 1.** Performanța unei soluții realizate de noi pentru task-ul 1 pe datele de validare atinge un scor precizie-medie de 0.798.

Protocolul de evaluare și măsura de performanță

Cuantificăm performanța algoritmilor voștri pentru task-ul 1 de detectare facială și task-ul 2 de recunoaștere facială în imagini test prin grafice de tip precizie-recall. Aceste grafice combină două valori:

- *precizia*: procentul de detectii returnate de algoritmul vostru ca fiind corecte (ele conțin o față de interes). În cazul ideal, algoritmul vostru are o precizie = 1 = 100%, adică fiecare detectie furnizată de algoritm reprezintă o față de interes. Pentru task-ul 1 toate fețele din imagine sunt fețe de interes, pentru task-ul 2 numai fețele unui personaj specific (Fred, Daphne, Shaggy sau Velma) sunt de interes.
- *recall* (= rată de detectare): procentul de fețe de interes din imaginile test localizate corect. În cazul ideal algoritmul vostru are un recall = 1 = 100%, adică localizează corect toate fețele de interes din imagine.

Fiecare punct de pe graficul precizie-recall reprezintă precizia și recall-ul algoritmului vostru obținute pentru toate detectiile (ordonate descrescător după scor) care depășesc un anumit scor prag (threshold). De aceea, este foarte important ca atât pentru task-ul 1 cât și pentru task-ul 2 să aveți un scor asociat detectiilor voastre. Sumarizăm întregul grafic prin *precizia medie* care reprezintă aria de sub grafic. Funcțiile *eval_detections* (pentru task-ul 1) și *eval_detections_character* (pentru task-ul 2) realizează aceste grafice și calculează precizia medie. Este important de reținut că fără un scor asociat pentru detectii nu putem calcula aria de sub graficul precizie-recall întrucât în acest caz graficul va fi reprezentat de un singur punct.

Figura 4 ilustrează graficul precizie-recall și precizia medie = 0.789 pentru task-ul 1 pe setul de date de validare obținută de una din soluțiile noastre. Un detector facial perfect ar avea precizia medie = 1, cel mai din dreapta punct de pe graficul funcției aflădu-se în acest caz în colțul din dreapta sus.

Figura 5 ilustrează graficele precizie-recall pentru fiecare personaj în parte pentru task-ul 2 pe setul de date de validare obținute de una din soluțiile noastre.

Restricții în implementarea soluției

Implementarea voastră pentru soluția de bază trebuie să urmeze paradigma de glisare a unei ferestre (sliding-window), extragerea de caracteristici pentru fiecare fereastră și folosirea unui clasificator. Puteti folosi o paletă întreagă de caracteristici: histograme de gradienți orientați, caracteristici obținute folosind rețele convolutionale, caracteristici bazate pe culoare, etc.

Nu puteti folosi pentru soluția de bază algoritmi de tipul Faster-RCNN sau YOLO sau rețele pre-antrenate pe fețe care v-ar ușura rezolvarea problemei. Îi puteti folosi pentru a lua puncte bonus. Dacă există neclarități ne puteti întreba dacă aveți sau nu voie să folosiți anumite lucruri.

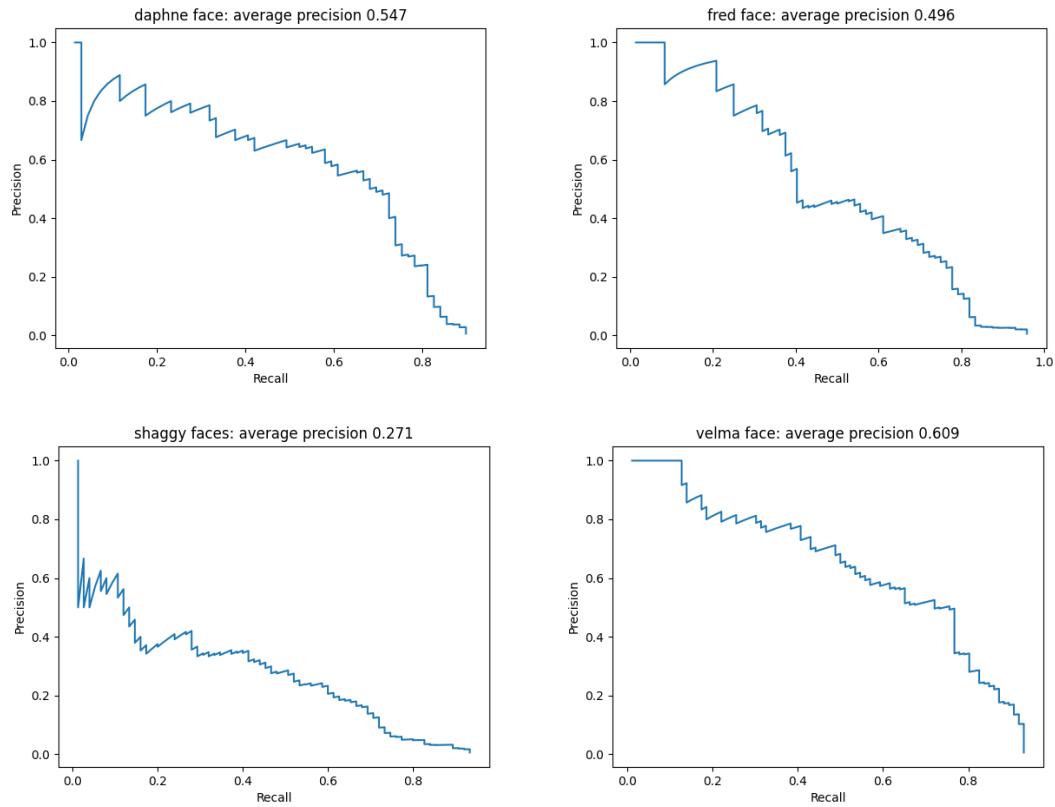


Figura 5: **Grafic precizie-recall pentru task-ul 2.** Performanța unei soluții realizate de noi pentru task-ul 2 pe datele de validare atinge un scor precizie-medie de 0.547 pentru Daphne, 0.496 pentru Fred, 0.271 pentru Shaggy și 0.609 pentru Velma.

Notare

Pentru această temă, vom folosi următoarele reguli de notare:

- **Task 1 - 4 puncte** - vom evalua performanța algoritmului vostru în problema detectării faciale pe o mulțime de 200 imagini de testare. Vom folosi ca măsură de performanță precizie-medie (Average Precision = AP) obținută pe baza graficului precizie-recall. Vom calcula punctajul corespunzător soluției voastre astfel:
 - o soluție care atinge un prag de $x\%$ AP va obține $\frac{4*x}{80}$ puncte, dar nu mai mult de 4 puncte. Practic acordăm 4 puncte pentru soluțiile care ating pragul de 80% AP.
 - soluția propusă de noi (Figura 4) ar obține $\frac{4*79.8}{80} = 3.99$ puncte (pe mulțimea de validare).
 - o soluție care atinge un prag de 40.0% AP va obține 2 puncte.
 - o soluție care depășește pragul de 80% AP va obține 4 puncte, nu se iau puncte bonus aici.

- **Task 2 - 4 puncte** - vom evalua performanța algoritmului vostru în problema recunoașterii faciale pe aceeași mulțime de 200 imagini de testare. Vom folosi ca măsură de performanță media preciziilor-medii (mean Average Precision = mAP) obținută pe baza celor patru grafice precizie-recall. Vom calcula punctajul corespunzător soluției voastre astfel:
 - o soluție care atinge un prag de $x\%$ mAP va obține $\frac{4*x}{60}$ puncte, dar nu mai mult de 4 puncte. Practic acordăm 4 puncte pentru soluțiile care ating pragul de 60% mAP.
 - soluția propusă de noi ar obține un $mAP = (0.547 + 0.496 + 0.271 + 0.609) / 4 = 0.480$ (Figura 5) care ar însemna 3.20 puncte.
 - o soluție care atinge un prag de 30% mAP va obține 2 puncte.
 - o soluție care depășește pragul de 60% mAP va obține 4 puncte, nu se iau puncte bonus aici.
- **BONUS - 25% din punctajul pentru fiecare task** - vă puteți crește punctajul cu 25% pentru fiecare task, dacă folositi o soluție secundară (un detector *state-of-the-art* de obiecte/fețe) de tip Faster-RCNN sau YOLO, pe care să o adaptați la problema voastră și a cărei performanță să depășească pragul impus pentru fiecare task (80%AP pentru task-ul 1 și 60% mAP pentru task-ul 2). Astfel, dacă obțineti cu soluția de bază 2 puncte la un task, puteți crește punctajul la 2,5 puncte condiționat de faptul că adaptați un detector pentru task-ul vostru și reușiți să depășiți pragul menționat.
- **documentație - 0,5 puncte** - descrieți într-un fisier pdf de minim două pagini soluția voastră de bază pentru rezolvarea celor două task-uri. Puteți ilustra aspecte cheie ale soluției voastre adăugând secvențe de cod și vizualizări ale imaginilor pentru soluția voastră. Acest fișier ar trebui să conțină suficientă informație astfel încât un student de nivel mediu de la cursul nostru să poată reimplementa soluția descrisă de voi. Dacă folositi și o soluție secundară pentru bonus includeți detalii desprea aceasta în documentație.
- **format - 0,5 puncte** - primiți 0,5 puncte dacă formatul fișierelor voastre urmează formatul impus iar codul nostru de evaluare rulează pe datele primite de la voi fără a face modificări în fișierele voastre. **Foarte important: soluția voastră trebuie să salveze fișiere .npy: pentru task-ul 1 aceastea sunt detections_all_faces.npy, scores_all_faces.npy, file_names.all_faces.npy iar pentru task-ul 2 aceastea sunt detections_character.npy, scores_character.npy și file_names_character.npy, care să poată fi încărcate direct cu np.load(...) de către scriptul evaluateaza_solutie.py. Fișierele trebuie să conțină variabilele detections, scores și file_names, cu dimensiuni și tipuri compatibile.** Verificați în acest sens directorul *evaluare*.
- **oficiu - 1 punct.**

Termene limită

Prima fază - trimiterea codului. Încărcați o arhiva zip cu codul soluției voastre și un fișier pdf ce descrie soluția voastră până duminică, 18 ianuarie 2026, ora 23:59 la link-ul acesta <https://tinyurl.com/CAVA-2025-TEMA2-SOLUTII>. Codul vostru ar trebui să includă un fișier README (vedeți exemplul din materiale) cu următoarele informații: (i) librăriile folosite de voi și necesare pentru rularea soluției voastre; (ii) indicații despre cum ar trebui rulat codul pentru fiecare task. Studenții care nu încarcă un fișier pdf cu descrierea soluției lor vor avea nota scăzută cu 0,5 puncte. Termenul limită este strict, nu vom accepta soluții primite mai târziu. **Implementarea voastră trebuie să fie gata de rulat pe setul de testare, astfel includeți toate fișierele ce conțin modele, descriptori etc.** În arhiva voastră pentru a putea fi încărcate direct. De asemenea, la fel ca și la Tema 1, nu acceptăm modificarea codului după trimiterea lui. Asigurați-vă că arhiva trimisă conține un cod sursă care poate fi rulat într-un environment nou, că aveți toate fișierele incluse și că parametri aleși sunt robusti și generalizează pe datele de validare. Sub nicio formă nu aveți voie să vă modificați parametri după trimiterea soluției.

A doua fază - trimiterea rezultatelor. Luni, 19 ianuarie 2026, vom publica datele de test în directorul *testare* de la adresa <https://tinyurl.com/CAVA-2025-TEMA2>. Veți rula soluția voastră pe imaginile de test și veți încărca rezultatele în aceeași zi ca o arhiva zip folosind următorul link <https://tinyurl.com/CAVA-2025-TEMA2-REZULTATE> în formatul recomandat.