

Detecția și recunoașterea fețelor utilizând un smartphone

Laurențiu-Ionuț Moloman ¹

dept. CTI, Grupa 1306B

Universitatea Tehnică "Gheorghe Asachi" din Iași
Iași, România

laurentiu-ionut.moloman@student.tuiasi.ro

Eduard-Constantin Minea ²

dept. CTI, Grupa 1306B

Universitatea Tehnică "Gheorghe Asachi" din Iași
Iași, România

eduard-constantin.minea@academic.tuiasi.ro

Ștefan-Daniel Achirei ³

dept. CTI

Universitatea Tehnică "Gheorghe Asachi" din Iași
Iași, România

stefan-daniel.achirei@academic.tuiasi.ro

Vasile-Ion Manta ⁴

dept. CTI

Universitatea Tehnică "Gheorghe Asachi" din Iași
Iași, România

vasile-ion.manta@academic.tuiasi.ro

Abstract—Proiectul curent se axează pe dezvoltarea și implementarea unui sistem de detectare și recunoaștere a fețelor utilizând dispozitive mobile, cu accent pe smartphone-uri. Detectarea și recunoașterea facială reprezintă aspecte esențiale în domeniul tehnologiei moderne, având aplicații variate în securitate, autentificare și interacțiunea cu dispozitivele.

Index Terms—face detection, face recognition, ML Kit, CameraX

[1] INTRODUCERE

Detectarea și recunoașterea fețelor sunt domenii de cercetare în continuă dezvoltare, având aplicații extinse în securitate, tehnologia informației și domenii conexe. În cadrul acestui proiect, ne propunem să explorăm fezabilitatea implementării unei soluții de detectare și recunoaștere a fețelor folosind un smartphone.

[1.1] Context

Odată cu avansarea tehnologiei smartphone-urilor, capacitățile acestora au crescut semnificativ. În prezent, aceste dispozitive sunt dotate cu camere puternice și tehnologii avansate de prelucrare a imaginilor, deschizând noi oportunități pentru dezvoltarea de aplicații inteligente de detecție și recunoaștere a fețelor.

[1.2] Obiectivele Proiectului

Scopul principal al proiectului nostru este să implementăm o soluție eficientă și precisă pentru detectarea și recunoașterea fețelor folosind un smartphone. Vom explora tehnologii de ultimă oră în domeniul învățării automate și prelucrării de imagini pentru a realiza o aplicație robustă și ușor de utilizat.

[2] STATE-OF-THE-ART

• Referința 1:

- Redimensionare (Resizing):

Cartea explorează conceptele fundamentale ale redimensionării imaginilor, care implică schimbarea

dimensiunilor unei imagini păstrând conținutul său vizual. Diverse tehnici de interpolare, cum ar fi cea de vecinătate cea mai apropiată (nearest-neighbor), biliniară și bicubică, sunt probabil discutate pentru a înțelege cum se interpolează pixelii în timpul procesului de redimensionare. Considerații referitoare la rezoluția spațială și metodele de anti-aliasing pot fi explorate pentru a asigura imagini redimensionate de înaltă calitate.

- Decupare (Cropping):

Decuparea implică selectarea unei regiuni specifice de interes în cadrul unei imagini, eliminând porțiunile nefolositoare. Se presupune că sunt acoperite tehnici de definire și manipulare a regiunilor de decupare, inclusiv considerații pentru menținerea raportului de aspect și selectarea dimensiunilor optime de decupare. Discuțiile pot include impactul decupării asupra conținutului imaginii, rezoluției și reprezentării vizuale generale.

- Rotație (Rotation):

Se explorează rotația imaginilor, acoperind atât tehnici de bază, cât și avansate. Transformările geometrice, cum ar fi matricele de rotație și transformările afine, pot fi discutate pentru a furniza o bază matematică pentru rotația imaginii. Cartea ar putea aborda, de asemenea, probleme legate de interpolare în timpul rotației și considerații pentru minimizarea pierderii de informație.

• Referința 2:

- Capitolul 6 - "Image Processing": discută despre conversia de date YUV.

Rusydi Matura discută despre conversia de date YUV în secțiunea "Formate de date de imagine" din capitolul 6. El începe prin a explica ce este formatul

YUV și de ce este utilizat în prelucrarea imaginilor. Apoi, el discută despre diferitele formate YUV, inclusiv YUV_420_888, YUV_42021 și YUV_42012.

În cele din urmă, el oferă un exemplu de cod care arată cum se poate converti datele YUV din formatul YUV_420_888 în formatul NV21. Acest cod este similar cu codul pe care l-ai menționat.

Iată un citat din secțiunea *"Formate de date de imagine"* din capitolul 6:

"Unul dintre formatele de date de imagine cel mai frecvent utilizate este YUV. Formatul YUV este o reprezentare a imaginilor colorate care separă luminanța (Y) de croma (U și V). Luminanța reprezintă nivelul de lumină al unei imagini, în timp ce croma reprezintă culoarea."

Formatul YUV este utilizat în prelucrarea imaginilor deoarece este eficient pentru stocarea și transmiterea datelor de imagine. De asemenea, este ușor de convertit în alte formate de date de imagine, cum ar fi RGB."

- Referința 3:

- *Concepte explicate:* Google ML Kit, învățare automată, viziune computerizată

- *Descriere:* Această carte este o resursă oficială de la Google care oferă o prezentare generală cuprinzătoare a Google ML Kit. Cartea acoperă toate aspectele importante ale SDK-ului, de la instalare și configurare până la utilizarea funcțiilor de învățare automată. De asemenea, cartea include exemple de cod și exerciții practice care ajută cititorii să învețe cum să folosească Google ML Kit în aplicațiile lor Android.

Cartea este o resursă valoroasă pentru dezvoltatorii Android care doresc să învețe cum să utilizeze TensorFlow Lite pentru a dezvolta aplicații de învățare automată. Cartea acoperă o gamă largă de subiecte, inclusiv conceptele de bază ale învățării automate, introducerea în TensorFlow Lite, pregătirea datelor pentru învățarea automată, construirea modelelor de învățare automată, implementarea modelelor de învățare automată în aplicațiile Android și utilizarea hardware-ului accelerat pentru învățarea automată.

- Referința 7:

- *"Android Machine Learning with TensorFlow Lite"* este o carte scrisă de Raja Kannan ce oferă o introducere completă la utilizarea TensorFlow Lite pentru dezvoltarea de aplicații de învățare automată pentru Android. Acoperă o gamă largă de subiecte,

inclusiv conceptele de bază ale învățării automate, introducerea în TensorFlow Lite, pregătirea datelor pentru învățarea automată, construirea modelelor de învățare automată, implementarea modelelor de învățare automată în aplicațiile Android și utilizarea hardware-ului accelerat pentru învățarea automată.

- Referința 10:

- *"Face detection, classification and transformation android application"* scrisă de Anum Haroon și Sadaf Iqbal oferă o introducere la dezvoltarea de aplicații Android care detectează, clasifică și transformă fețele.

Cartea începe cu o introducere în conceptele de bază ale detectării feței, inclusiv cum să identificați fețele în imagini și videoclipuri. De asemenea, acoperă diferite tipuri de algoritmi de detectare a feței, cum ar fi algoritmi bazeți pe caracteristici și algoritmi bazeți pe învățarea automată.

În cele din urmă, cartea acoperă transformarea fețelor, care este procesul de modificare a aspectului unei fețe. Cartea acoperă diferite tipuri de transformări faciale, cum ar fi schimbarea expresiei faciale, schimbarea vârstei etc.

[3] SOLUȚII COMERCIALE

- *Microsoft Azure Face API:*

- *Descriere:* Microsoft Azure Face API oferă funcționalități puternice de detectare a fețelor, recunoaștere facială și analiză a expresiilor. API-ul poate fi utilizat pentru construirea de aplicații de securitate, gestionarea identității sau îmbunătățirea experienței utilizatorilor în aplicații mobile.

- *Amazon Rekognition:*

- *Descriere:* Amazon Rekognition este un serviciu de analiză a imaginilor oferit de Amazon Web Services. Acesta include funcții puternice de analiză facială, recunoaștere facială și detecție a expresiilor faciale. Amazon Rekognition poate fi utilizat pentru o gamă largă de aplicații, inclusiv securitate, publicitate și organizarea fotografiilor.

- *Google Cloud Vision API:*

- *Descriere:* Google Cloud Vision API oferă servicii avansate de analiză a imaginilor, inclusiv detecție facială și recunoaștere a fețelor. Acesta poate fi integrat în diverse aplicații pentru a efectua analize detaliate ale imaginilor și pentru a extrage informații relevante.

[4] METODOLOGIA DE CERCETARE

Pentru a atinge obiectivele proiectului nostru de detectare și recunoaștere a fețelor utilizând un smartphone, am adoptat o abordare riguroasă și bine structurată în ceea ce privește metodologia de cercetare. Procesul nostru de dezvoltare și evaluare a constat în următoarele etape:

[4.1] Studiu de Literatură:

Am efectuat o cercetare a literaturii de specialitate pentru a înțelege tehnologiile și metodele existente în domeniul detectării și recunoașterii fețelor. Am analizat lucrări științifice, articole de cercetare și resurse tehnice relevante pentru a identifica cele mai recente tendințe și inovații în acest domeniu.

[4.2] Selectarea Tehnologiilor:

Pe baza informațiilor obținute din studiul de literatură, am selectat tehnologiile potrivite pentru implementarea sistemului nostru. Am ales să utilizăm TensorFlow Lite și ML Kit pentru a beneficia de avantajele învățării automate și pentru a asigura eficiență în implementarea pe dispozitive mobile.

[4.3] Integrarea în Aplicația Android:

Pasul următor a constat în integrarea modelului de detecție facială într-o aplicație Android. Am utilizat Android Studio pentru dezvoltarea aplicației, asigurându-ne că interacțiunea cu funcționalitățile de detecție și recunoaștere facială este fluentă și prietenoasă pentru utilizator.

[4.4] Evaluarea Performanțelor:

Am evaluat performanțele sistemului nostru în diverse scenarii de utilizare. Am măsurat acuratețea detecției, viteza de procesare și eficiența în condiții variate, ajustând și optimizând modelul și codul sursă conform necesităților.

Prin implementarea acestei metodologii, ne propunem să obținem un sistem robust, eficient și ușor de utilizat pentru detectarea și recunoașterea fețelor pe dispozitivele mobile, cu o atenție deosebită acordată performanțelor și experienței utilizatorului.

[5] REZULTATE INTERMEDIARE

[5.1] Performanța Detectorului de Față:

Detectorul de față furnizat de Google ML Kit a demonstrat o performanță remarcabilă în detectarea fețelor în timp real, oferind rezultate precise și consistente pe dispozitive Android.

[5.2] Preprocesarea Imaginilor:

Etapele de preprocesare, inclusiv transformarea imaginilor YUV în bitmap și ajustarea orientării, au contribuit semnificativ la asigurarea calității datelor de intrare pentru modelul de recunoaștere facială.

[5.3] Manipularea Imaginilor Salvate:

Implementarea funcțiilor de salvare și încărcare a fețelor a contribuit la gestionarea eficientă a datelor, permițând utilizatorilor să își gestioneze seturile de fețe recunoscute.

[6] DISCUȚII ȘI CONCLUZII PARȚIALE

În această secțiune, vom examina rezultatele parțiale ale proiectului nostru și vom aborda aspecte relevante în evoluția acestuia. Progresul până în acest moment a adus la lumină mai multe observații și provocări semnificative.

[6.1] Rezultate Parțiale

Până în acest moment, am obținut rezultate promițătoare în ceea ce privește detectarea și recunoașterea fețelor utilizând smartphone-uri. Algoritmii noștri au demonstrat capacități semnificative, iar interfața cu utilizatorul a fost receptată pozitiv în teste preliminare.

[6.2] Provocări Întâmpinate

Cu toate acestea, ne-am confruntat și cu anumite provocări care necesită atenție suplimentară. Problemele legate de iluminare variabilă și calitatea camerei pot afecta performanța sistemului nostru. De asemenea, am identificat zone în care algoritmii noștri pot fi îmbunătățiți pentru o acuratețe sporită.

[6.3] Lecții Învățate

Pe parcursul acestui proiect, am învățat importanța testării riguroase și adaptării continue. Rezultatele evaluărilor ulterioare au fost resurse valoroase pentru ajustarea soluției noastre și înțelegerea mai profundă a nevoilor utilizatorilor.

[6.4] Ajustări Viitoare

Pe baza rezultatelor vom explora noi abordări pentru a răspunde provocărilor identificate. Acest lucru poate include optimizări ale algoritmilor noștri, adaptarea la medii cu iluminare variabilă și îmbunătățirea generală a experienței utilizatorului.

Această discuție parțială oferă o imagine cuprinzătoare asupra stadiului proiectului nostru și a direcției viitoare pe care intenționăm să o urmărim. Concluziile finale vor fi elaborate în secțiunea următoare, după finalizarea proiectului.

[7] DIRECȚII VIITOARE

Proiectul nostru reprezintă doar începutul unei explorări a domeniului detectării și recunoașterii fețelor utilizând smartphone-uri. În viitor, intenționăm să ne concentrăm asupra următoarelor aspecte pentru a îmbunătăți și extinde soluția noastră:

[7.1] *Îmbunătățiri ale Preciziei*

Vom explora metode avansate de prelucrare a imaginilor și algoritmi de învățare automată pentru a îmbunătăți precizia detectării și recunoașterii fețelor. Implementarea unor tehnici de învățare profundă sau adaptarea modelelor existente ar putea aduce beneficii semnificative în acest sens.

[7.1] *Dezvoltarea unei Interfețe Utilizator Îmbunătățite*

Vom investi în dezvoltarea unei interfețe utilizator intuitive și prietenoase, oferind utilizatorilor o experiență plăcută și acces facil la funcționalitățile aplicației noastre.

[7.4] *Extinderea la Mai Multe Platforme*

Vom evalua posibilitatea extinderii aplicației noastre pentru a suporta mai multe platforme mobile, inclusiv iOS și altele. Această extindere va crește accesibilitatea soluției noastre pentru o gamă mai largă de utilizatori.

REFERENCES

- [1] Gonzalez and Woods, "Digital Image Processing".
- [2] Rusydi Matura, "Java Programming for Computer Vision: From Beginner to Advanced".
- [3] Vasco Correia Veloso , "Hands-On Artificial Intelligence for Android: Understand Machine Learning and Unleash the Power of TensorFlow in Android Applications with Google ML Kit".
- [4] Dawn Griffiths and David Griffiths, "Head First Android Development".
- [5] John Omokore, "Android Camera2 API".
- [6] Bharath Ramsundar and Reza Bosagh Zadeh, "TensorFlow for Deep Learning: From Linear Regression to Reinforcement Learning".
- [7] Raja Kannan, "Android Machine Learning with TensorFlow Lite".
- [8] Bill Phillips and Chris Stewart, "Android Programming: The Big Nerd Ranch Guide".
- [9] Brian Goetz, "Java Concurrency in Practice".
- [10] Anum Haroon, Sadaf Iqbal, "Face detection,classification and transformation android application", July 28, 2013.
- [11] C. Ismael, "List of 10+ Face Detection / Recognition APIs, libraries, and software," 19 June 2013.
- [12] Li Ma, Lei Gu and Jin Wang, "Research and Development of Mobile Application for Android Platform", Vol. 9, No.4, pp.187-198, (2014).
- [13] Ex-Sight, "Face Recognition - Technology Overview", 2009.