

Oportunități de utilizare a calculelor distribuite în prelucrarea imaginilor

Mircea Petic, Inga Țițhiev, Grigorii Horoș

Institutul de Matematică și Informatică al AȘM

inga.titchiev@gmail.com, petic.mircea@gmail.com, grigorii.horos@gmail.com

Abstract — articolul conține descrierea cercetării posibilităților de procesare a imaginilor la nivel de recunoaștere a formelor, și anume a fețelor umane, folosind tehnici de calcul distribuit. Lucrarea conține o trecere în revistă a definițiilor și noțiunilor ce țin de sisteme distribuite și legătura cu domeniul de procesarea imaginilor. Este descris procesul de recunoaștere a formelor. Aplicația este prezentată la nivel de schemă logică, algoritm și tehnologiile folosite. Valoarea aplicației este argumentată cu rezultatele obținute în urma rulării ei.

Index Terms — sistem distribuit, procesarea imaginilor, recunoașterea formelor, hare-like features, openCV.

I. INTRODUCERE

În perspectivă atât europeană cât și globală tinerii cercetători din Republica Moldova simt necesitatea de a dobândi abilități și competențe pentru a fi competitivi și pentru a contribui la dezvoltarea societății. Una dintre cele mai recente direcțiile dezvoltate în cadrul Institutului de Matematică și Informatică al Academiei de Științe din Republica Moldova este utilizarea infrastructurilor de calcul de înaltă performanță pentru prelucrare distribuită a datelor în timp real la rezolvarea diverselor probleme [4]. Fiind încă un domeniu destul de nou, dar împreună cu o anumită experiență în procesare de imagini și dezvoltarea infrastructurii de calcul de înaltă performanță, domeniul devine o oportunitate care oferă multe avantaje la rezolvarea problemelor din diverse domenii.

Pentru a rămâne competitivi pe piața tehnologiilor informaționale, în ultimele decenii, numărul oamenilor de știință și al companiilor ce se bazează pe sisteme de calcul distribuite a crescut semnificativ. Deoarece costurile mentenanței unor super-computere proprii sînt foarte ridicate, nu toate companiile sau grupurile de cercetare își permit costurile generate de achiziția infrastructurii necesare rulării propriilor aplicații sau experimente. Astfel aceștia trebuie să apeleze la resurse provenite din surse externe [3].

În acest context scopul acestui articol constă în cercetarea posibilităților de procesare a imaginilor la nivel de recunoaștere a formelor folosind tehnici de calcul distribuit.

Din aceste considerente articolul este structurat în felul următor. Inițial se face o trecere în revistă a definițiilor și noțiunilor ce țin de sisteme distribuite și legătura acestora cu domeniul de procesare a imaginilor. În continuare se descrie procesul de recunoaștere a formelor, și anume a fețelor umane folosind algoritmul de clasificare. Aplicația realizată este descrisă la nivel de schemă logică, algoritm (Haar-like features) și tehnologiile folosite (OpenCV, node.js). Pe final sînt prezentate rezultatele obținute în urma rulării aplicației cu unele concluzii și problemele depistate în urma aplicării algoritmului de recunoaștere a obiectelor pe imagini.

II. SISTEME DISTRIBUITE

Prin sistem distribuit de calcul sau sistem informatic distribuit se înțelege o mulțime de programe peste o rețea de noduri (calculatoare, multiprocesoare, procesoare paralele masive, stații de lucru, clustere, grid, etc.) care au acces fiecare la o memorie proprie (dar pot avea acces și la anumite memorii comune partajate), fiind conectate între ele prin niște linii de comunicație (fir, fibră optică, unde radio, sateliți), avînd diverse topologii de conexiune (magistrală comună, stea, etc.), sistemul fiind conceput cu scopul partajării unor resurse sau/și pentru rezolvarea concurentă a unor aplicații paralele sau paralelizabile [5].

Procesarea imaginilor (adică pre-procesarea, recunoașterea formelor, identificarea unor particularități specifice), reprezintă un caz particular de utilizare al sistemelor și tehnologiilor de procesare distribuită a informației. Rezolvarea acestei probleme necesită folosirea sistemelor distribuite deoarece de cele mai multe ori aceasta necesită resurse de calcul mari.

Pentru echilibrarea încărcării, se poate realiza aplicația astfel încît aceasta să ruleze pe mai multe servere. Și anume această abordare va fi folosită la dezvoltarea acestui proiect.

Se va utiliza un server numit serverul principal, cu capacitatea de 1Gb RAM, un alt server va fi folosit pentru stocarea de fișiere, cu capacitatea de 1Gb RAM și 60Gb spațiu liber, și 3 noduri pentru recunoaștere de imagine, cu capacitatea de 2Gb RAM. Acest lucru îi va permite aplicației să fie foarte rapidă și fiabilă.

III. PROCESAREA IMAGINILOR

Procesarea imaginilor reprezintă un punct de interes comun pentru mai multe comunități științifice în vederea reducerii timpului de procesare și proiectare a unor algoritmi optimizați de prelucrare [8]. Are la bază o teorie matematică riguroasă, dar implementările deseori sînt destul de mari și consumatoare de resurse (putere de calcul, memorie), în special cînd este vorba despre utilizarea în timp real a informațiilor extrase din imagini, după cum s-a

menționat și mai sus pentru a putea face față necesităților date și se vor utiliza oportunitățile oferite de către sistemele distribuite.

Prelucrarea și analiza imaginilor cuprinde ansamblul de tehnici și metode de achiziție, stocare, afișare, modificare și exploatare a informației vizuale cuprinsă în imagini. În particular, analiza imaginilor se referă la capacitatea de a descrie, înțelege și recunoaște scene, obiecte din scene și legăturile dintre acestea.

Domeniul prelucrării imaginilor este unul foarte dinamic, și în această lucrare vom utiliza sintetizarea făcută de către Pavlidis în [1].

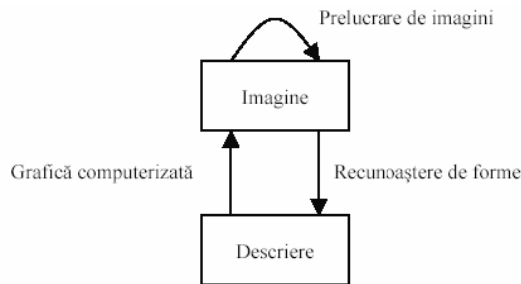


Figura 1. Domeniul prelucrării imaginilor

Luând în considerare schema din Figura 1, ce reprezintă domeniul de prelucrare a imaginilor, ținem să precizăm că în continuare ne vom concentra asupra recunoașterii de forme.

IV. RECUNOAȘTEREA FORMELOR

Recunoașterea formelor [2] reprezintă o modalitate de a extrage informații din imaginile achiziționate. Este un domeniu larg, ce include: recunoașterea scrisului, recunoașterea feței umane, recunoașterea amprentelor etc. Recunoașterea formelor constă într-o clasificare și/sau o descriere a conținutului imaginii.

În acest articol vom utiliza *clasificarea* care constă în atribuirea unei forme necunoscute din imagine la o clasă dintr-un set predefinit de clase. După aplicarea operației de clasificare vom obține la ieșire o imagine nouă care reprezintă o *hartă* a obiectelor determinate.

Algoritmii de clasificare se bazează pe extragerea caracteristicilor (trăsăturilor) pe o măsură a similarității (de exemplu distanța). Pentru selectarea caracteristicilor este necesară o cantitate destul de mare de informație.

Un clasificator conține trei module:

1. modulul de clasificare propriu zis;
2. modulul de învățare (presupune prezența unui set de imagini de antrenare);
3. modulul de selecție și extragere a caracteristicilor.

Toate aceste module sînt implementate în limbajul C++ ca părți componente a aplicației finale.

V. SCHEMA APLICAȚIEI

Pentru a ilustra funcționarea sistemului de calcul distribuit bazat pe recunoașterea de forme a fost elaborată o aplicație Web care recunoaște fețele umane și care oferă următoarele funcționalități:

- Încărcarea imaginilor pe server (este permisă în

regim multiutilizator)

- Stocarea imaginilor pe un server de fișiere
- Recunoașterea imaginilor (este permisă de asemenea în regim multiutilizator), care se efectuează cu ajutorul unui modul elaborat în limbajul C++ utilizînd biblioteca OpenCV
- Ștergerea imaginilor de pe server

Aplicația elaborată utilizează un sistem distribuit de calcul care oferă facilități de utilizare eficientă a resurselor disponibile. Schema funcțională a aplicației elaborate este prezentată în Figura 2.

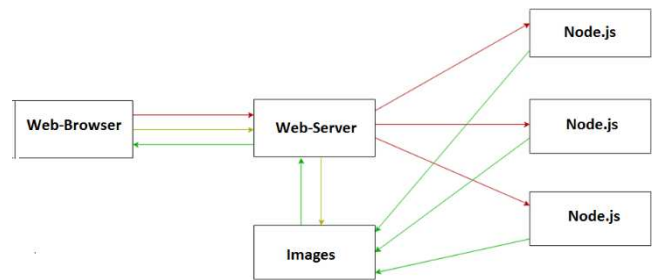


Figura 2. Schema funcțională a aplicației elaborate

Prezența în Figura 2 a liniilor roșii denotă faptul că prin această operație imaginile sînt încărcate pe server nefiind încă procesate.

Operația indicată prin linie verde semnifică faptul că la acel moment deja imaginile au fost prelucrate.

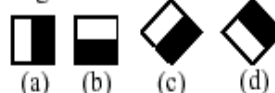
Operațiile reprezentate prin linii galbene determină cererea imaginilor prelucrate de către browser.

Calculul distribuit utilizat în aplicație este argumentat prin faptul că utilizarea celor trei noduri va permite efectuarea rapidă și eficientă a calculului necesare recunoașterii formelor.

VI. HAAR-LIKE FEATURES

Pentru a putea recunoaște un obiect din imagine (în particular față) se va utiliza algoritmul Haar-like featurescare permite acest lucru. La baza recunoașterii respective după cum este indicat și în Figura 3, se utilizează dreptunghiuri cu culori alb și negru care pot avea diferite poziții și mărimi pe imagine.

1. Edge features



2. Line features



3. Center-surround features



Figura 3. Elemente de recunoaștere a formelor

Recunoașterea se face în următorul mod: Dreptunghiul cu trăsăturile Haar se deplasează deasupra imaginii care se procesază. Pentru fiecare poziție a dreptunghiului se calculează suma conform formulei de mai jos:

$$\sum_{i=0}^{W-1} \sum_{j=0}^{H-1} I(x+i, y+j) - I(i, j),$$

unde W și H sînt dimensiunile imaginii inițiale, i și j – reprezintă poziția dreptunghiului cu trăsături Haar, locul unde suma data este minimală și determină poziția obiectului căutat [6].

VII. OPENCV

Pentru a realiza implementarea algoritmului Haar-like features s-a ales drept instrument librăria OpenCV care oferă posibilități de realizarea ale acestuia.

OpenCV este o librărie open source de recunoașterea imaginilor inițial creată de către Intel. Această librărie este scrisă în C și C++, și are module pentru Python, Ruby, Matlab și alte limbaje. Ea este gratuită pentru uz comercial deoarece are licența BSD. Această librărie este cross-platform: lucrează în Mac OS X, Windows și Linux.

Utilizînd OpenCV au fost implementate următoarele funcționalități:

- Încărcarea imaginilor utilizînd funcția:
`cv::imread()`
- Lucrul cu fișiere `.xml` asupra cărora se aplică algoritmul Haarcascades utilizînd funcția:
`cv::CascadeClassifier::load()`
- Recunoașterea imaginilor utilizînd funcția `cv::CascadeClassifier::detectMultiScale()`
- Numărarea obiectelor de pe imagine.

VIII. NODE.JS

Pentru a putea utiliza aplicația on-line oferindu-i posibilitate de access de oriunde și oricînd s-a folosit instrumentarul oferit de către Node.js.

Node.js este o platforma pentru rularea programelor scrise în JavaScript pe server. El este cross-platform, și lucrează cu cele mai cunoscute sisteme de operare: Linux, Windows și MacOS [9]. Oportunitățile utilizării NodeJS constau în:

- Lucru ușor cu fluxuri multiple
- Viteza de lucru a aplicației este mai înaltă datorită faptului că NodeJS utilizează motorul de rulare al JavaScript-ului V8, care de asemenea este folosit și în browser-ul Google Chrome [7].

IX. EVALUARE

Pentru a verifica eficiența utilizării acestei aplicații a fost utilizat un set de imagini de diferite dimensiuni care corespund formatului `.jpeg`.

În Figura 4 este prezentat un rezultat al rulării aplicației pe o imagine care conține mai multe fețe. Din imagine se observă că toate fețele au fost detectate, fiind încercuite. În cazul acesta este vorba de o imagine de o calitate bună care nu prezintă momente mai complicate pentru algoritmul la

detectare.



Figura 4. Rezultatul execuției aplicației pentru o imagine

Datele concrete în urma rulării aplicației sînt prezentate în Tabelul 1. Se observă că în medie au fost folosite imagini care conțin în mediu 3-4 fețe.

TABEL I. REZULTATELE ÎN LUCRU ALE APLICAȚIEI

Caracteristica	Valoarea
Numărul de imagini (buc.)	102
Numărul de fețe total (buc.)	390
Numărul mediu de fețe pe o imagine	3.82
Numărul de fețe găsite (buc.)	329
Numărul de fețe găsite acolo unde nu erau	80
Numărul de fețe acolo unde erau	51
Timpu de execuție (s)	390

Se observă faptul că aplicația nu recunoaște toate fețele care sînt în imagini. Au fost găsite în plus fețe acolo unde nu erau și nu au fost găsite fețe acolo unde ele erau.

În urma analizei rezultatelor au fost scoase în evidență problemele care nu permit recunoașterea obiectelor din imagini. Acestea sînt următoarele:

- Probleme cu luminozitatea și contrast
- Nu se depistează fețele care sunt sub un unghi de deviație mai mare de 30 grade.
- OpenCV nu recunoaște imagini cu dimensiuni mici

X. CONCLUZII

Prelucrarea imaginilor este un domeniu complex și foarte dinamic, cu numeroase aplicații în diverse domenii (biologie, medicină, etc.). În prezent există un șir de algoritmi ce permit extragerea informațiilor utile din imagine. Scopul acestui articol a fost prezentarea implementării ce utilizează calculele distribuite ca factor de

eficientizare a resurselor utilizate în prelucrarea imaginilor.

Aceste rezultate sunt un prim pas în determinarea unui algoritmul care va putea efectua ulterior și o căutare a întregii imagini dintr-o bază de date. Acestea vor putea fi utilizate în particular la căutarea în bazele de date cu imagini medicale a unor particularități specifice unei patologii, sau depistarea unei patologii după imaginea dată.

ACKNOWLEDGMENTS

Acest articol a fost elaborat în cadrul proiectului pentru tineri cercetători: 13.819.18.06A Sisteme și tehnologii de procesare distribuită a informației și evaluarea eficienței de utilizare a acestora.

REFERENCES

- [1] T. Pavlidis, Algorithms for Graphics and Image Processing, Computer Science Press, 1982.
- [2] C. M. Mark, "Metode de prelucrare a imaginilor" Anale. Seria Informatică. Vol I fascII, pp 188-204, Universitatea "Tibiscus", Timisoara, 2003.
- [3] Gabriela Andreea Morar, Utilizarea resurselor de calcul distribuite pentru o execuție eficientă a aplicațiilor de business complexe. Rezumat teză de doctorat, 2012, 43 p
- [4] I. Țițchiev, M. Petic, N.Iliuha, Opportunities for young researchers from Republic of Moldova, Proceedings of the Third International Conference on «INFORMATICS AND COMPUTER TECHNICS PROBLEMS» (PICT – 2014), 27 – 30 May, 2014 Chernivtsi, Ukraine, pp. 197-198.
- [5] I. Dzițac, G. Moldovan, Sisteme distribuite modele informatice, Editura Universității Agora, Oradea, 2006, 146 p.
- [6] Gary Bradski and Adrian Kaehler, Learning OpenCV, O'Reilly Media, 2008, pp 524-526
- [7] R. Laganière. Computer Vision Application Programming Cookbook. BIRMINGHAM – MUMBAI. First published: May 2011. ISBN 978-1-849513-24-1
- [8] I. Țițchiev, M. Petic, V. Macari, Posibilități de procesare paralelă ale imaginilor prin intermediul funcțiilor OPENCV, Conferința internațională științifico-practică "Dezvoltarea inovațională din Republica Moldova: problemele naționale și tendințele globale", 7-8 noiembrie, pp.589-591, 2013, ISBN 978-9975-4266-0-2.
- [9] M. Cantelon, T. J. Holowaychuk, M. Harter, N. Rajlich, Node.js in Action, Manning Publications Company, 2013, 395 p.