

Hand Detection using a Set of Classifiers Based on Haar-like Features

Elena-Gina CRĂCIUN, Ștefan-Gheorghe PENTIUC

Abstract—Humans hands are very important in interaction with the environment. Many different hand poses can be assumed to manipulate objects or convey information. Therefore the human hand can serve as input device for a computer, much more then simply moving a mouse in a 2D plane. Think of manipulating objects in a virtual world or using gestures to communicate. To obtain this information several approaches exist. Besides using a glove with sensors or using a camera and markers on the hand it is also possible to approach the problem from a purely computer vision based side. This has the desirable possibility of being non-obtrusive for the user. This is however complicated due to the articulate nature of the hand and a possibly complex and changing environment.

In this paper will try to describe the solution for the problem of real-time vision based hand detection. The lower level of the approach analyzes the posture recognition with Haar-like features and the AdaBoost learning algorithm. With this algorithm, real-time performance and high recognition accuracy can be obtained. The higher level for this problem is to obtain a solution to resolve the hand gesture recognition using different methods such as a context-free grammar-based syntactic analysis, Hidden Markov Model or Sparse Bayesian Classifier.

Index Terms—AdaBoost, gesture, hand, Haar-like features, OpenCV, posture, recognition

I. INTRODUCERE

Interfețele om-calculator au evoluat vertiginos în decursul ultimilor ani, dovadă fiind implementarea unor sisteme virtuale în care interacțiunea om-calculator este realizată cât mai natural. Aceste sisteme oferă în prezent noi tehnici de comunicare, interacțiune, învățare și antrenare cu ajutorul utilizatorului. Dispozitivele de comunicare tradiționale 2D, tastatura și mouse, au devenit inutile în noul context și astfel s-a simțit nevoia utilizării unor modalități și tehnologii diferite. Dispozitivele care detectează poziția corpului și recunosc gesturile, vorbirea, expresiile feței și alte aspecte ale comportamentului uman pot fi folosite pentru o comunicare om-mediul virtual mult mai eficientă și interactivă [1].

Interacțiunea om-calculator din cadrul mediilor virtuale consideră mâna dispozitiv de intrare. *Gesturile umane* sunt un instrument puternic de comunicare între oameni; cu toate acestea expresivitatea gesturilor umane nu a fost suficient cercetată ca și tehnică de explorare.

Mâna umană este un obiect complex și foarte articulat. Dacă se ia în considerare poziția mâinii și fiecare deget, mișcare realizată de acesta are aproximativ 27 de grade de libertate (DOF). Pentru folosirea mâinii ca și un dispozitiv natural HCI, s-au folosit dispozitive asemănătoare cu mănușile, CyberGlove, pentru captarea mișcării acesteia.

Inconveniente care au apărut în acest caz nu au fost puține; se pot menționa următoarele: mănușile și firele atașate acestora erau incomode pentru utilizator, prețul ridicat a unui astfel de dispozitiv pentru utilizatori obișnuiți. Ultimele cercetări în domeniul computer – vision au arătat că recunoașterea mâinii și clasificarea gesturilor în timp real este metoda cea mai convenabilă pentru interacțiunea om-computer în realitatea virtuală.

În cadrul procesului de recunoaștere a mâinii apar anumite *limitări* cum ar fi: *forma* acesteia se schimbă foarte rapid odată cu mișcarea; schimbarea nu se realizează doar pentru postura mâinii, dar fiecare postură este diferită din unghiuri diferite; fiecare deget are 3 încheieturi, deci 4 DoF pentru fiecare deget. Degetul mare este și mai flexibil și are 5 DoF. Astfel configurația doar pentru degete are 21 DoF. În combinație cu orientarea globală și poziție rezultă un total de 27 DoF. Toate aceste lucruri sunt importante și sunt necesare pe durata procesului de recunoaștere. Pe lângă *numărul mare de posibile variații* în timpul mișcării pentru o mână, sunt diferențe semnificative care apar de la om la om: *forma, culoarea, mărimea* (ex. mâna unui copil diferă foarte mult de mâna unui adult bărbat). În plus anumite părți ale mâinii pot fi acoperite de alte părți (ex. degetele se pot acoperi unele pe altele). *Ocluzia* unei zone apare și prin acoperirea unei mâini de către cealaltă mână sau de diverse obiecte din scenă. Acesta face imposibilă observarea posturii complete pentru mână [2].

Unele probleme depind și de anumite condiții specifice aplicației. În cazul recunoașterii gesturilor, pot apărea diferite inconveniente cum ar fi poziția mâinii, care se schimbă de la un frame la altul și astfel mișcarea mâinilor devine foarte confuză.

Atunci când pentru recunoașterea gesturilor este capturat întreg corpul, mâinile ocupă doar o mică parte din imagine la un nivel foarte redus. Acesta face mult mai dificilă distincția mâinii față de celelalte componente.

Pentru a fi un sistem performant, nu se poate impune restricții pentru background. Astfel celelalte obiecte care se află static sau în mișcare în fundal ar trebui tolerate. În particular, în cazul sistemelor pentru recunoașterea limbajului mimico-gestual, mâinile sunt poziționate în dreptul feței și uneori ating fața. Din acest motiv este dificil să facem ipoteze bazate de exemplu pe diferența de culoare dintre mâinii și background.

Este foarte posibil ca pe durata unui gest, mâinile să se atingă sau chiar să se unească. Acesta nu va duce doar la ascunderea unei zone pentru o mână dar vor fi obținute și configurații greșite ale mâinilor [3].

II. CARACTERISTICILE HAAR ȘI ADABOOST

Pentru utilizarea gesturilor în cadrul sistemelor virtuale, sistemul de recunoaștere trebuie să îndeplinească anumite cerințe în ceea ce privește acuratețea, performanța în timp real și robustețea. Procesul de recunoaștere într-un mediu aglomerat este punctul cheie pentru un astfel de sistem. Un astfel de proces utilizează înaintea algoritmi care se bazează pe culoarea pielii sau forma mâinii. Atunci când se dorea eliminarea problemelor care apăreau ca urmare a faptului că mâna poate avea aceeași culoare cu fața sau cu alte obiecte din cadru, utilizatorul era obligat să poarte tricouri lungi sau se impuneau anumite restricții asupra obiectelor din zona de lucru. O altă problemă care apare în cazul algoritmilor care se bazează pe culoare este sensibilitatea ridicată la variația luminii. În ceea ce privește algoritmi care au la bază forma mâinii aceștia folosesc descriptori pentru forma acestora [4]. Descriptorii au la bază prelucrarea pixelilor, operație costisitoare pentru sistemele de timp real. Dezavantajul algoritmilor bazați pe forma mâinii este necesitatea de a „curăța” imaginea segmentată, filtrare, lucru greu de realizat pentru imaginile cu un background aglomerat și afectat de zgomot.

Pentru a veni în întâmpinarea acestor probleme sistemele actuale folosesc un set de caracteristici Haar care au fost folosite cu succes în cazul detecției feței. Fiecare caracteristică Haar poate fi privită ca un template de mai multe dreptunghiuri albe și negre interconectate.

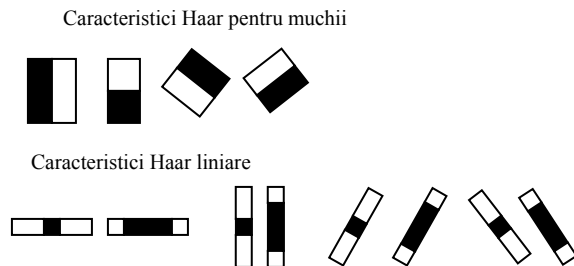


Fig. 1. Caracteristici Haar.

Există două motive pentru care se dorește implementarea cu ajutorul caracteristicilor Haar și nu cu valorile pixelilor neprelucrate. Primul motiv ar fi acela că trăsăturile Haar pot codifica domeniul de cunoaștere, lucru care este destul de dificil de realizat folosind o cantitate finită de date de antrenare. Comparativ cu pixeli neprelucrați, caracteristicile Haar pot determina cu ușurință apartenența sau nu la clasa de antrenare și astfel realizarea clasificatorului este realizată mult mai simplu. Trăsăturile Haar descriu și raportul dintre zona închisă și luminoasă din imaginea antrenată. Un exemplu tipic este regiunea din zona ochiului uman este mult mai închisă decât regiunea obrazurilor. Al doilea motiv este acela că un sistem care are la bază caracteristicile Haar poate lucra la o viteză mult mai ridicată decât un sistem bazat pe prelucrarea pixelilor. În afară de aceste două avantaje, caracteristicile Haar sunt puternice la schimbarea condițiilor de lumină și zgomot

deoarece ele compun nivelul de gri prin diferența dintre dreptunghiurile albe și negre.

Valoarea caracteristicii Haar este diferența dintre suma pixelilor din dreptunghiul negru și suma pixelilor din dreptunghiul alb.

$$f(x) = W_{black} \cdot \sum_{black_region} (pixel_value) - W_{white} \cdot \sum_{white_region} (pixel_value),$$

W_{black} și W_{white} sunt valori care îndeplinesc condiția compensației:

$$W_{black} \cdot black_region = W_{white} \cdot white_region \quad [5]$$

Pentru detectarea obiectului de interes, imaginea este scanată cu ajutorul unei sub-ferește cu o anumită caracteristică Haar. Pe baza fiecărei caracteristici f_i , se definește un clasificator corespunzător $h_j(x)$ astfel:

$$h_j(x) = \begin{cases} 1, & p_j f_j(x) < p_j \theta_j \\ 0 \end{cases}$$

unde x este sub-fereastra, θ este pragul, p_j indică direcția.

În practică detecția precisă a unui obiect nu se poate realiza cu ajutorul unei singure caracteristici. Oricum, se poate obține cu ușurință o serie de clasificatori slabi, cu o precizie aproape peste 50% folosind caracteristicile Haar.

Boosting este o metodă iterativă care încurajează modelele noi să devină modele „expert” pentru instanțele clasificate greșit de modelele anterioare. Justificarea intuitivă ar fi aceea că modelele trebuie să acționeze ca și experți care se completează unul pe altul [6].

- se construiește o secvență de clasificatori $C_1, C_2, \dots, C_n$
- C_1 este construit pe setul de antrenare
- C_2 acordă mai multă atenție instanțelor clasificate greșit de C_1
- C_3 acordă mai multă atenție instanțelor clasificate greșit de C_1 și C_2

Algoritmul adaptabil de învățare AdaBoost, introdus prima dată de către Freund și Schapire în [7] a rezolvat anumite dificultăți care erau în algoritmi boost și are rolul de a îmbunătăți precizia clasificatorilor slabi odată cu creșterea numărului de stagii. În practică, implementarea acestei cascade are rolul de a crește performanța algoritmului de învățare. În primul stadiu al procesului de antrenare, pragul clasificatorului slab este ajustat foarte jos astfel încât obiectele țintă să poată fi detectate în procent de 100%. Pentru a putea fi detectat de cascada de antrenare, sub-fereastra pozitivă trebuie să treacă prin fiecare stadiu al cascadei. Un rezultat negativ în orice punct conduce la o respingere a acelei sub-ferește.

Motivul aplicării acestei strategii se bazează pe faptul că majoritatea sub-fereștrilor sunt negative în cadrul unui singur frame, și este puțin probabil ca o fereastră pozitivă să parcurgă

toate stagiile. Cu ajutorul acestei strategii, cascada poate crește considerabil timpul de procesare întrucât inițial clasificatorul slab va încerca să respingă cât mai multe subferestre negative și resursele computaționale câștigate sunt folosite pentru clasificarea acelor imagini care au trecut de analiza stagiilor inițiale ale cascadei.

Algoritmul AdaBoost distribuie ponderile uniform pentru fiecare set de antrenare. Este aleasă caracteristica Haar care a realizat o clasificare cu o performanță ridicată la prima iterație. Clasificatorul slab care are la bază această caracteristică este adăugat setului de clasificatori cu un parametru proporțional cu performanța sa.

La următoarea iterație ponderile pentru seturile de antrenare sunt re-eșantionate: imaginile din setul de antrenare care nu au fost clasificate de primul clasificator slab sunt favorizate astfel încât a doua caracteristică Haar să acorde mai multă atenție acestora. Pentru a fi selectată, a doua caracteristică Haar trebuie să obțină o performanță mai ridicată pentru aceste seturi neclasificate și astfel eroarea globală să fie redusă. Aceste iterații se repetă prin adăugarea de noi clasificatori la setul total până când se atinge o performanța globală cât mai bună. Rezultatul de antrenare final este un clasificator puternic compus dintr-un set de clasificatori slabi.

III. DETECȚIA MÂINII

În cadrul unui sistem pentru detecția mâinii sunt necesare parcurgerea mai multor etape, așa cum se poate observa în Fig. 2. Achiziția imaginilor s-a realizat cu ajutorul unei camere web. Colecția obținută pentru setul de antrenare este formată din imagini negative și imagini pozitive. Rezoluția imaginilor a fost de 640x480. Imaginile negative sunt imagini arbitrare, din diferite medii și care nu conțin obiectele pe care dorim să le detectăm. Imaginile pozitive sunt imagini care conțin obiectele pentru detecție; acestea sunt marcate pentru a putea fi utilizate de clasificator în procesul de antrenare.

Pasul de învățare – training – se realizează cu ajutorul algoritmului AdaBoost; fereastra de scanare a fost de 20x20 cu un număr de 30 de stagii. Setul de antrenare împreună cu o familie de clasificatori „slabi” construiesc un clasificator puternic salvat sub forma unei cascade într-un fișier xml. Clasificatorul astfel obținut este utilizat în aplicația care testează sistemul.

În cadrul etapei de testare imaginile folosite sunt diferite față de cele pe care s-a realizat procesul de învățare. Cu ajutorul funcțiilor din librăria OpenCV imaginile au fost supuse următorului algoritm:

1. extragerea zonei de interes din imagine (zona care conține obiectul detectat)
2. conversia imaginii în nuanțe de gri
3. binarizarea imaginii după un anumit prag (threshold)
4. obținerea conturului pentru obiectul detectat

IV. CONCLUZII ȘI DIRECȚII DE DEZVOLTARE

În această lucrare s-a încercat o sumară introducere în realizarea unui sistem ce permite detecția unui obiect, mâna în acest caz. Cu ajutorul caracteristicilor Haar și a algoritmului AdaBoost sistemul poate detecta posturi ale mâinii în timp real, în condiții luminoase variate. Imaginile sunt preluate cu ajutorul unei camere. Pe viitor se dorește îmbunătățirea clasificatorului, prin colectarea imaginilor de la un număr ridicat de persoane diferite, în condiții unui background aleator; astfel acest sistem va deveni mult mai independent și robust. Următorul pas este recunoașterea gesturilor și integrarea acestei aplicații într-un sistem virtual care să permită realizarea unor operații diferite: navigare în medii 3D sau operații de asamblare. O altă îmbunătățire ar fi realizarea unui sistem care să realizeze simultan atât recunoașterea posturii și a gesturilor dar și a altor obiecte cum ar fi fața umană, ochii. Relațiile dintre aceste obiecte pot avea înțelesuri diferite și astfel se poate crea un set nou de comenzi care pot fi integrate în alte sisteme virtuale bazate pe interacțiunea cu utilizatorul. Pentru o experiență mult mai interactivă se pot integra și alte tehnici de comunicare cum ar fi recunoașterea vocii.

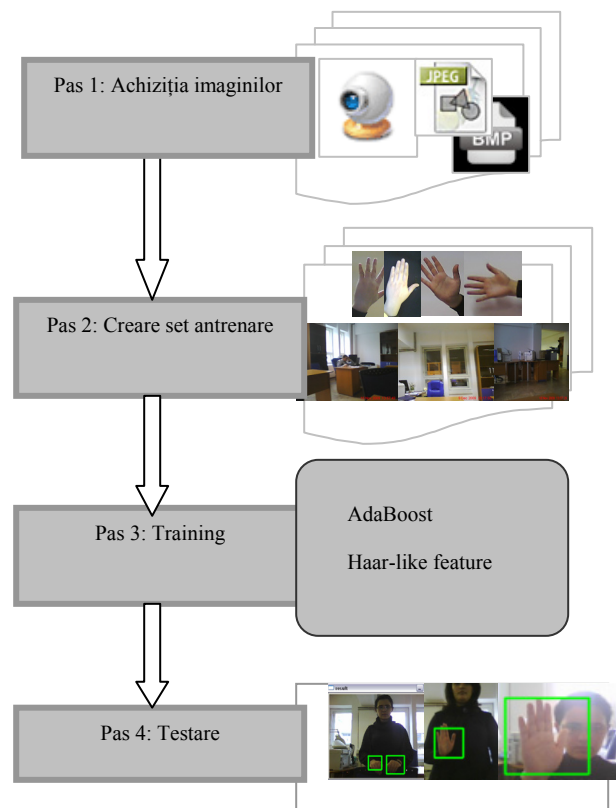


Fig. 2. Arhitectura sistemului.

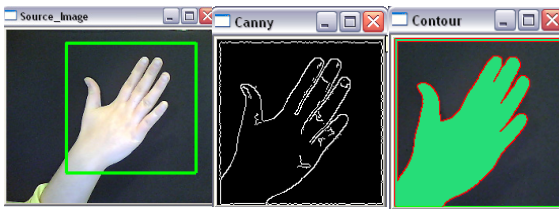


Fig. 3. Detectie obiect.

REFERENCES

- [1] Prateem C., Prashant S., Ankit M., "Hand Gesture Recognition – A Comparative Study", in Proceedings of the International Multi Conference of Engineers and Computer Scientists 2008 Vol I IMECS 2008, 19-21 March, 2008, Hong Kong.
- [2] M.K. Bhuyan, P.K. Bora, and D. Ghosh, "Trajectory Guided Recognition of Hand Gestures having only Global Motions", International Journal of Computer Science, Fall 2008.
- [3] Atid Shamaie Alistair Sutherland, "Accurate Recognition of Large Number of Hand Gestures", Centre for Digital Video Processing School of Computer Applications, Dublin City University, Dublin, 2005
- [4] Q. Chen, F. Malric, Yi Zhang, M. Abid, "Interacting with Digital Signage Using Hand Gestures", Institute of Computer Graphics and Image Processing, Tianjin University, China, 2008.
- [5] Jacob Whitehill, Christian W. Omlin, "Haar Features for FACS AU Recognition", Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition, p: 97 – 101, 2006.
<http://www.econ.ubbcluj.ro/~gheorghe.silaghi/datamining2008/curs9.pdf>
- [6] Freund, Y., Schapire, R. E., "A short introduction to boosting", Journal of Japanese Society for Artificial Intelligence, vol. 14, no. 5, pp. 771-780 (1999).
- [7] http://www.hotbanners.com/print-icons/jpeg_icon.jpg
- [8] http://dryicons.com/images/icon_sets/coquette_part_4_icons_set/png/128x128/bmp_file.png
- [9] Afshin Sepehri, Yaser Yacoob, Larry S. Davis "Employing the Hand as an Interface Device", Journal of Multimedia, vol 1, number 2, pages 18-29
- [10] Ivan Laptev and Tony Lindeberg "Tracking of Multi-state Hand Models Using Particle Filtering and a Hierarchy of Multi-scale Image Features", Proceedings Scale-Space and Morphology in Computer Vision, Volume 2106 of Springer Lecture Notes in Computer Science, pages 63-74, Vancouver, BC, 1999. ISBN 978-3-540-42317-1, doi 10.1007/3-540-47778-0
- [11] Lars Bretzner, Ivan Laptev, Tony Lindeberg "Hand gesture recognition using multi-scale colour features, hierarchical models and particle filtering", Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition, Washington, DC, USA, 21-21 May 2002, pages 423-428. ISBN 0-7695-1602-5, doi 10.1109/AFGR.2002.1004190
- [12] Thomas B. Moeslund and Lau Nørgaard, "A Brief Overview of Hand Gestures used in Wearable Human Computer Interfaces", Technical report: CVMT 03-02, ISSN: 1601-3646, Laboratory of Computer Vision and Media Technology, Aalborg University, Denmark.
- [13] Stenger B, Thayananthan A, Torr PH, Cipolla R: "Model-based hand tracking using a hierarchical Bayesian filter", IEEE Transactions on Pattern Analysis and Machine Intelligence, 28(9):1372-84, Sep 2006.
- [14] A Erol, G Bebis, M Nicolescu, RD Boyle, X Twombly, "Vision-based hand pose estimation: A review", Computer Vision and Image Understanding Volume 108, Issues 1-2, October-November 2007, Pages 52-73 Special Issue on Vision for Human-Computer Interaction, doi:10.1016/j.cviu.2006.10.012.
- [15] New Xbox: Full body and gesture recognition (Project Natal), http://www.erwinvanlun.com/ww/full/new_xbox_full_body_gesture_recognition/
- [16] Ying Wu and Thomas S. Huang, "Vision-Based Gesture Recognition: A Review", In: Gesture-Based Communication in Human-Computer Interaction, Volume 1739 of Springer Lecture Notes in Computer Science, pages 103-115, 1999, ISBN 978-3-540-66935-7, doi 10.1007/3-540-46616-9
- [17] Open Computer Vision Library, OpenCV Framework (<http://sourceforge.net/projects/opencvlibrary>), OpenCV wiki (<http://opencv.willowgarage.com/wiki/>)
- [18] Bradski, G.; Kaehler, A. (2008), Learning OpenCV: Computer Vision with the OpenCV Library, O'Reilly and Partners (<http://www.amazon.com/Learning-OpenCV-Computer-Vision-Library/dp/0596516134>)
- [19] Beat Signer, Moira C. Norrie and Ueli Kurmann - iGesture: A Java Framework for the Development and Deployment of Stroke-Based Online Gesture Recognition Algorithms, Technical Report ETH Zurich, TR561, September 2007.