

Urmărirea degetelor mâinii prin tehnici de viziune pe calculator pentru utilizare în interacțiunea om-calculator

Bradi Rares

Universitatea Tehnică Cluj-Napoca
Str. Memorandumului nr. 28, CP 400114
bradi.rares@gmail.com

Dorian Gorgan

Universitatea Tehnică din Cluj-Napoca
Str. Memorandumului nr. 28, CP 400114
dorian.gorgan@cs.utcluj.ro

REZUMAT

Sarcina de recunoaștere a gesturilor mâinii este una activ cercetată și provocatoare din cauza fundalurilor complexe, medii cu iluminări diferite și mișcări variate ale mâinii. Lucrarea de față prezintă o implementare și analiză a unui sistem capabil să recunoască centrul mâinii, degetele, precum și încheietura acesteia. Recunoașterea centrului mâinii se face prin cercul de rază maximă înscris în conturul ei, iar vârful degetelor utilizând curbura-k. Traectoria mâinii este urmărită și diferitele poziții intermediare sunt analizate, pentru a identifica o direcție de deplasare. Pentru a avea robustețe și stabilitate se face urmărirea trăsăturilor prin filtre Kalman. Gesturile sunt compuse dintr-o direcție de deplasare și o anumită poziționare a vârfulor degetelor. Evaluarea sistemului urmărește robustețea la recunoaștere, viteza de procesare și rata de eroare.

Cuvinte cheie

Recunoaștere vizuală pe calculator, Interacțiune om-calculator, segmentare piele, gest dinamic, urmărire mână.

Clasificare ACM

H5.2. Information interfaces and presentation: Input devices and strategies.

INTRODUCERE

Una dintre barierele existente în interacțiunea om-calculator este modalitatea prin care interacționăm cu fluxul mare de informație [4]. Soluția este dezvoltarea unor tehnici de interacțiune “naturală” cu calculatorul, asemănătoare celor prin care oamenii interacționează între ei. Folosirea gesturilor mâinii reprezintă o alternativă naturală la tehnicile de interacțiune din interfețele utilizator clasice. Prin formele non-verbale de comunicare, realizate prin mișcarea mâinilor, gesturile devin componente ale unui limbaj de dialog cu calculatorul, utilizat pentru schimbul de informație.

Dezvoltarea tehnicilor de recunoaștere vizuală pe calculator, face posibilă abordarea recunoașterii gesturilor din perspectiva comunicării dintre om și calculator.

Conform lucrării [5] cele două caracteristici importante în proiectarea unei aplicații interactive sunt utilizabilitatea și funcționalitatea. Pentru a realiza o interacțiune om-calculator performantă și de bună calitate, este necesar să se realizeze un balans între aceste două caracteristici [5].

Lucrarea de față abordează recunoașterea gesturilor mâinii centrată pe cerințele de funcționalitate și utilizabilitate.

Dificultățile care apar în implementarea unor tehnici de recunoaștere centrate pe aceste cerințe sunt:

- Detecția prezenței și a poziției mâinii în scenă;
- Identificarea clasei de gesturi căreia îi aparține forma mâinii din scenă;
- Determinarea deplasării în scenă a obiectului identificat;
- Latența sistemului;
- Stabilitatea urmăririi: variația poziției măsurate să fie cât mai mică, dacă obiectul măsurat nu se mișcă.

Detecția prezenței mâinii în scenă se face prin determinarea cercului de rază maximă care poate fi potrivit în interiorul conturului mâinii. Identificarea gesturilor se face prin intermediul vârfulor degetelor extrase folosind curbura-k ale conturului mâinii care îndeplinesc condițiile necesare. Pe baza acestor determinări se identifică degetul mare și cel mic, ca fiind acele vârfulor care satisfac condițiile de poziționare relativă în regiunea mâinii. Urmărirea se face într-o fereastră de analiză de dimensiune fixă, care acumulează date precedente și ia decizii pe baza lor. Pentru a avea stabilitate în datele captate se folosesc filtre Kalman. Astfel se elimină variațiile bruște ale datelor măsurate și permite o estimare bună a poziției, chiar în momentele în care măsurătorile lipsesc. Traectoria urmărită este clasificată pe baza punctelor intermediare captate, utilizând predicția liniară.

În secțiunea Realizări Asemănătoare sunt prezentate alte sisteme construite, folosind tehnici și algoritmi diferiți, dar cu același rezultat final pentru utilizator. În Proiectarea Arhitecturii se prezintă principalele componente funcționale. În Extragerea Trăsăturilor se analizează în detaliu algoritmi folosiți în detecția principalelor caracteristici prin care se face identificarea gestului. Secțiunea Recunoașterea Gesturilor descrie soluțiile problemelor de urmărire și recunoaștere. În secțiunea Evaluare și Rezultate sunt descrise scenariile de test relevante și indicatorii de performanță folosiți. Ultima parte, Concluzii evidențiază pe scurt ideile și observațiile importante expuse în lucrare.

REALIZĂRI ASEMĂNĂTOARE

Ca urmare a activităților de cercetare în domeniul recunoașterii gesturilor care pot opera în timp real, s-au conturat clase de abordări distincte. O primă clasă sunt cele care folosesc modelul matematic/computațional de rețele neuronale artificiale. Acestea își propun simularea structurii biologice a sistemului neuronal al omului. Rețelele neuronale acceptă ca intrare trăsături și produc la

ieșire decizii. În lucrarea [8] autorii aplică acest model pentru recunoașterea limbajului de semne realizate cu mâinile. Se folosește conceptul de perceptron ca algoritm de clasificare. Trăsăturile alese pentru antrenarea lui sunt histogrammele orientării gradientilor, în care operatorul Sorbel, din pașii de calcul a trăsăturilor a fost înlocuit printr-un alt operator, însă având coeficienți similari. Această abordare presupune antrenarea modelului, într-o fază de preprocesare, înaintea rulării. Față de metoda programată folosită în abordarea noastră, soluția rețelei neuronale bazate pe antrenare oferă avantajul adaptării la condiții care se schimbă într-un ritm constant. Metoda programării poate eșua pentru situații care diferă de cele luate în calcul în etapa proiectării. Totuși formarea bazei de date și antrenarea modelului sunt consumatoare de timp și pot fi repetate doar în anumite condiții. Un alt dezavantaj este că sistemul recunoaște doar gesturi statice și nu are mijloace prin care să analizeze comportamente în timp.

Lucrarea [8] captează datele vizuale despre poziția mâinii folosind o cameră Kinect. Segmentarea mâinii, folosind informațiile de adâncime, determină conturul mâinii să conțină mai puține goluri în interior față de alte metode de segmentare. Din conturul extras, se aleg puncte echidistante pentru care se calculează distanța față de centroidul regiunii mâinii. Pe aceste puncte se aplică transformata Fourier, iar rezultatul obținut poartă denumirea de descriptori Fourier. Un gest este dat de o grupare de astfel de descriptori. La recunoaștere se aplică distanța euclidiană dintre modelul antrenat și cel captat, gestul recunoscut fiind cel pentru care se obține scorul cel mai mic. Principalul dezavantaj vine din presupunerea că mâna este cel mai apropiat obiect de interes din scenă, lucru care nu este valabil în toate cazurile. Abordarea prezentă nu se bazează pe această presupunere. O altă distincție față de lucrarea prezentată reprezintă trăsăturile folosite pentru a modela gesturile. Deoarece se folosesc vârful degetelor, modelul folosit în această lucrare are o dimensiune mică, iar recunoașterea are un cost computațional mai redus decât în [9].

O abordare diferită, folosind două tipuri de șabloane, este folosită în lucrarea [6] pentru determinarea vârfulor și clasificarea degetelor. Șablonul pentru deget este un cerc de rază mică, iar cel pentru centrul palmei, un cerc de rază mult mai mare. Cu aceste cercuri, se umple separat, pentru fiecare tip de șablon, imaginea segmentată astfel încât nici un cerc să nu părăsească regiunea de prim-plan segmentată. Se obțin astfel două imagini umplute cu cercuri de diferite raze. Fiecare cerc din șablonul palmei, este comparat cu cele din șablonul degetelor, pentru a nu se suprapune. Dacă se suprapun, cercul degetului este șters. Regiunea palmei reprezintă grupul format din cel mai mare număr de cercuri învecinate. Regiunile degetelor sunt reprezentate de grupurile de cercuri ale șablonului degetelor, care sunt învecinate cu regiunea palmei. Metoda este intens computațională. Față de aceasta algoritmul curburilor-k folosit în lucrarea prezentă este mai rapid.

În lucrarea [6] se prezintă o altă metodă de determinare a vârfulor degetelor, prin învelirea conturului mâinii într-o formă convexă și calcularea pe baza ei a defectelor de

convexitate care apar atunci când degetele sunt ridicate. Aceste regiuni sunt caracterizate de punctele de început și sfârșit de pe contur, precum și de distanța punctului minim din regiune la obiectul convex. Ca vârfuri sunt considerate punctele de început ale acelor regiuni care au o adâncime ce depășește un prag. Defectele de convexitate nu pot identifica vârfulurile de deget dacă ele sunt lipite. Algoritmul curburilor-k însă poate determina și în aceste condiții vârfulurile degetelor prin ajustarea parametrului k. Acest lucru face abordarea curentă mai robustă. De asemenea, în cadrul lucrării se face o recunoaștere a gesturilor dinamice pe când în [6] se recunosc doar gesturile statice.

PROIECTAREA ARHITECTURII SISTEMULUI

Arhitectura sistemului este prezentată în Figura 1.

La bază, sistemul propus, este format din patru componente: 1. Modulul de preluare a imaginilor de la camera video. Pe lângă rolul de captare el mai are sarcina de a scala în jos imaginea, pentru o procesare rapidă.

2. Modulul de detecție alcătuiește backend-ul sistemului. El îndeplinește rolul detecției trăsăturilor. Pentru a putea opera în condiții complexe, precum schimbări de iluminare, umbrii sau ocluziuni date de elemente din scenă, el comunică intern cu logica de urmărire și corecție pentru a putea identifica corect trăsăturile.

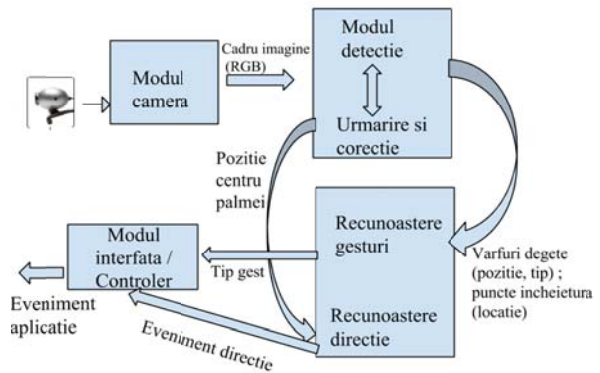


Figura 1. Arhitectura sistemului de recunoaștere a gesturilor mâinii

3. Modulul de recunoaștere. Odată identificate, trăsăturile mâinii sunt transmise componentei în care sunt modelate tipurile de posturi recunoscute și, componentei de recunoaștere a direcțiilor de deplasare. 4. Agregarea acestor informații în timp și efectuarea unei decizii finale ținând cont de rezultate în momente de timp anterioare, este sarcina ultimei componente de controler. Utilizând această informație ea va transmite evenimentul de producere a gestului aplicației utilizator.

Considerente tehnologice

Sistemul folosește în implementarea sa biblioteca OpenCV[14]. Aceasta abstractizează elementele care nu fac scopul acestui articol (modulul camerei) și pune la dispoziție un cadru împreună cu algoritmi de bază pentru prelucrarea imaginilor (ex: detecție contur, conversii medii de culoare).

EXTRAGEREA TRĂSĂTURILOR

Trăsăturile sunt cele care detectează prezența sau absența mâinii din cadrul de imagine. Ele se determină la nivelul modulului de detecție iar stagiile de procesare sunt detaliate în Figura 2.

Segmentarea regiunii mâinii

Scopul acestei etape este acela de a separa pixelii aparținând mâinii, de cei aparținând fundalului. Astfel se limitează zona asupra căreia celelalte etape vor acționa, iar cantitatea de procesat va fi redusă. Există mai multe tehnici care se pot aplica în această etapă, dar toate au drept caracteristică folosirea operațiilor simple.

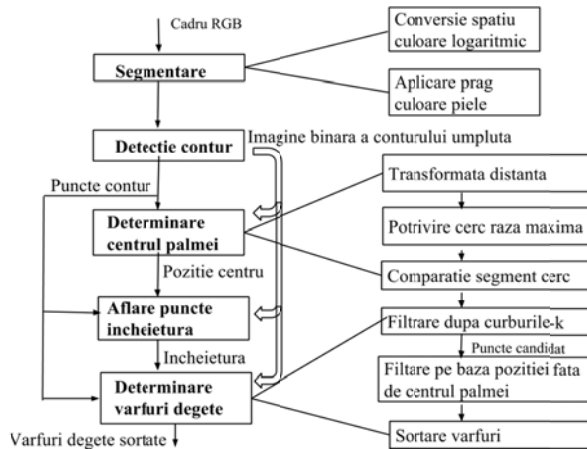


Figura 2. Etapele principale de procesare pentru extragerea trăsăturilor

În lucrare este folosită tehnica de segmentare bazată pe un interval de culoare. Fiecare pixel al cadrului este verificat dacă valoarea lui se află într-un interval specific culorii pielii. Avantajul major este simplitatea și aplicarea individuală la nivelul fiecărui cadru. Dezavantajele constau în schimb în faptul că diferențele de iluminare pot afecta culoarea obiectului dorit de analizat. Pentru a diminua influența pe care o are iluminatul asupra segmentării adesea se face o transformare într-un alt spațiu de culoare invariant la aceasta precum HSV sau $YCbCr$. Lucrări precum [3], [11] au studiat intervalul corespunzător pielii în astfel de spații, însă sunt lucrări [1] care propun noi spații de culoare mai robuste. O evaluare a acestora a fost făcută pe o bază de date de imagini statice împreună cu masca de adevăr [15].

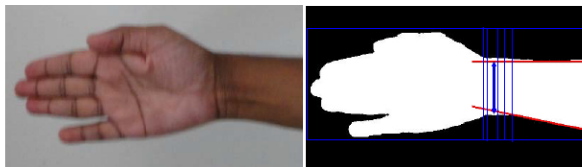


Figura 3. Imaginea originală în stânga iar în dreapta imaginea binară cu regiunea mâinii segmentată. Dreptele albastre reprezintă liniile de baleiaj, iar cea groasă candidatul pentru încheietură.

Pe baza rezultatelor s-a ales spațiul propus în [1] cu intervalul de culoare sugerat de autori $\log(R/G)=[0.15, 1.1]$; $\log(B/G)=[-4, 0.3]$. Rezultatul aplicării segmentării asupra mâinii se poate vedea în Figura 3, imaginea din dreapta.

Detecție contur

În această etapă se determină un contur al regiunii segmentate format de punctele care-l compun. În unele cazuri regiunea mâinii poate prezenta găuri negre în interiorul conturului regiunii segmentate. Acesta este umplut pentru stagiul următor. Determinarea și umplerea se fac prin funcții ale bibliotecii OpenCV.

DETECȚIA CENTRULUI PALMEI

În lucrările referite [9],[6],[8], detecția și poziția mâinii sunt date de existența unui centru al palmei. O definiție des utilizată pentru centrul palmei este punctul dat de cercul de rază maximă care poate fi înscris în regiunea mâinii. Pentru determinarea razei maxime se aplică transformata distanță pe imaginea binară. Aceasta în implementarea curentă din OpenCV folosește algoritmul descris în [2] care calculează cea mai scurtă cale către un pixel al fundalului, folosind deplasări pe orizontală, verticală, diagonală sau deplasarea calului de șah. La ieșire vom avea o imagine în care valoarea fiecărui pixel reprezintă distanța anterior amintită. Centrul cercului va corespunde pixelului cu cea mai mare distanță, iar cea din urmă reprezintă raza. Din această cauză este necesară umplerea conturului mâinii.

Există însă excepții în care cercul de rază maximă nu reprezintă cercul de mâinii, precum imaginea din Figura 4.



Figura 4. Exemplu caz critic în care metoda directă ar alege primul cerc. Aplicând măsura putem diferenția astfel de situații. Segmentele de cerc figurează cu cyan

Soluția care se propune este folosirea unei măsuri secundare pentru a discerne astfel de excepții. Măsura este lungimea totală a segmentelor de piele aflate pe un cerc suprapus regiunii mâinii. Determinarea lungimii segmentelor de cerc se face prin algoritmul lui Bresenham pentru desenarea cercului [16].

Detecția încheieturii

Autorii în [7] studiază această problemă din perspectiva unei implementări în timp real și având o rată de eroare scăzută. Observația exprimată în lucrarea lor este regăsită și în alte abordări și spune că lățimea încheieturii formează un minim local în silueta mâinii [7]. Acest lucru se poate observa în Figura 3. Lucrarea folosește propria implementare pentru a determina acest minim. Algoritmul este descris informal în următorii pași aplicați Figurii 3:

1. Se încadrează silueta mâinii cu un dreptunghi care minimizează suprafața acoperită.
2. Se parcurge suprafața dreptunghiului cu o linie de baleiaj de-a lungul laturii mari a dreptunghiului. Linia

are direcția dată de latura mică și punctul de început și sfârșit se află pe latura mare (liniile subțiri albastre).

3. Pentru fiecare astfel de linie de baleiaj se calculează lungimea segmentului de piele (linia albastră groasă) și se rețin punctul de început și sfârșit al acestuia.
4. Se rețin acele segmente de minim local dintr-o fereastră de analiză de dimensiune fixă (în implementare 10).

Încheietura va reprezenta segmentul minim cu lungimea mai mare decât un prag minim și sub un prag maxim (raza centrului palmei). Dacă pentru o linie de baleiaj se găsesc mai multe astfel de segmente, atunci se alege segmentul de lungime maximă. Pentru eficiență, din analiza făcută prin linia de baleiaj se pot elimina extremitățile dreptunghiului, deoarece încheietura nu se află în acea zonă a mâinii.

Determinarea vârfurilor degetelor

Vârful unui deget poate fi privit ca o curbă sau cotitură de-a lungul conturului mâinii. Detecția vârfurilor degetelor se va face pe baza acestei observații, algoritmul fiind cunoscut în literatură drept k-curvature. Procedura de determinare a punctelor candidat pentru vârfuri este ilustrată pe Figura 5.

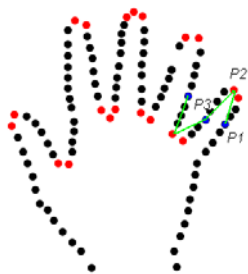


Figura 5. Ilustrarea calcului curburilor-k pentru $k = 4$. Punctele roșii sunt puncte candidat alese. Cele albastre sunt la distanță k față de cel analizat. Unghiul format de cele trei puncte este $P_1P_2P_3$

Fiecare punct de pe conturul mâinii este analizat. Pentru el se găsesc vecinii aflați la distanța k , după și înaintea lui în vector. Curbura k este definită ca unghiul format de aceste trei puncte în ordinea din Figura 5. Dacă unghiul este sub un anumit prag atunci acel punct este candidat a fi punct de vârf al degetului. Din figură se observă că există mulți candidați falși. În cele ce urmează se impun condiții care să-i elimine. Datorită anatomiei mâinii vârfurile degetelor se vor regăsi în pozițiile de pe contur aflate pe aceeași parte cu planul delimitat de încheietură, în care se află și centrul palmei. Astfel se elimină regiunea antebrațului din analiză. O altă problemă este filtrarea acelor candidați care se află într-un punct de vale de pe contur. Decizia se face asemănător cu observația de dinainte. Vârfurile se vor găsi în aceeași parte a planului delimitat de dreapta formată de vecinii săi distanțați k . Pentru vârfuri adiacente, cum regăsim în Figura 5, vom face o medie a candidaților. Un alt exemplu de candidați sunt încheieturile degetelor care respectă condiția de curbă. Aceștia sunt eliminați pe baza distanței euclidiene dintre poziția lor și poziția centrului palmei. Acele puncte

rămase în urma acestor trieri, vor fi recunoscute ca vârfuri de degete pentru sistem.

RECUNOAȘTEREA GESTURILOR

Recunoașterea gesturilor în sistemul nostru are trei componente. Prima componentă face recunoașterea unei posturi a mâinii definită de pozițiile degetelor, tipul degetelor și poziția centrului palmei într-un cadru de imagine. A doua componentă, urmărește traiectoria în timp a mâinii. Ea discerne cele patru direcții generale de deplasare: stânga, dreapta, sus, jos. Ultima componentă va combina aceste informații în timp pentru a putea decide starea unui gest.

Recunoașterea gesturilor statice

Deoarece se dorește diversitate în modelarea diferitelor posturi ale mâinii, sistemul va trebui să decidă și tipul fiecărui deget. În starea curentă, sistemul poate eticheta degetul mare și cel mic dacă le consideră prezente. Etichetarea se face pe baza a trei observații: 1. Dacă degetul mare este vizibil, atunci el va fi, fie cel mai din dreapta, fie cel mai din stânga. La fel este valabil și pentru degetul mic în funcție de mână și orientarea față de cameră; 2. Unghiul format de degetul mic sau degetul mare cu centrul palmei și centrul încheieturii sunt mai ascuțite decât cele formate de restul degetelor cu aceleași puncte; 3. Degetul mare este mai aproape de centrul palmei decât cel mic.

Pentru prima observație este necesar găsirea unei metode de ordonare a vârfurilor, odată ce poziția lor este cunoscută. Algoritmul implementat este cel descris în [9]. În urma aplicării sortării, vom avea o secvență ordonată de poziții ale vârfurilor degetelor de la stânga la dreapta. Doar primul și ultimul vârf sunt analizate. Dacă unghiul format de acestea cu centrul palmei și mijlocul încheieturii este mai mare decât 120° acesta este eliminat din analiză. Dacă în final rămânem cu două vârfuri, cel pentru care distanța dintre proiecția vârfului pe direcția mâini și centrul palmei, este mai mică va fi etichetat ca deget mare. Dacă însă rămânem doar cu un singur candidat, atunci distanța anterior menționată este comparată cu două praguri. Dacă este mai mic decât primul atunci este clasificat ca deget mare, dacă este mai mare decât al doilea este clasificat ca deget mic. Prin direcția mâinii înțelegem vectorul format de mijlocul încheieturii și centrul palmei.

În continuare se exemplifică condițiile ce trebuie satisfăcute pentru recunoașterea unui gest static pe siluetele din Figura 6.



Figura 6. Exemple de gesturi ce pot fi modelate. Începând de la stânga ele sunt numerotate cu 1 până la 9.

Pentru postura 1 trebuie să găsim un vârf etichetat ca deget mare. Postura 2 necesită existența a două vârfuri dintre care unul să fie degetul mare. Următoarea postură este caracterizată de prezența celor 5 degete. Vecina ei este definită doar de centrul palmei. Postura 5 este dată de aceleași condiții ca 2, plus condiția ca unghiul format de cele două vârfuri să fie sub 70° . Postura 6 este definită de prezența unui vârf neetichetat. Postura 7 necesită existența

a două vârfuri neetichetate, iar postura 8 are nevoie de trei. Ultima este formată din vârful degetului mic și încă unul. Limitări ale modelării programatice exemplificate sunt: posturile trebuie să poată fi distincte în termenii cunoscuți de sistem; mâna poate lua alte forme diferite, dar care satisfac cerințele modelate de clasificare.

Urmărirea trăsăturilor

Extragerea trăsăturilor nu este lipsită de erori. Elemente precum, condițiile de iluminare, ocluziunile și limitări ai algoritmilor, determină apariția abaterilor bruște ale pozițiilor trăsăturilor în anumite cadre. Pentru ca sistemul să fie robust împotriva acestor abateri se folosește filtrul Kalman[13]. Deși OpenCV oferă cadrul prin care se poate folosi acest filtru, parametrii lui trebuie inițializați. Figura 7 pune în prim plan principalii parametri.

$$\begin{aligned} stare = x_t &= \begin{pmatrix} pos_x \\ pos_y \\ v_x \\ v_y \end{pmatrix} \quad observatia = z_t = \begin{pmatrix} pos_x \\ pos_y \end{pmatrix} \quad (1) \\ x_{t+1} &= F \cdot x_t + B \cdot u_t + E_x \quad (2) \\ z_t &= H \cdot x_t + E_z \quad (3) \end{aligned}$$

Figura 7. Parametri filtrului Kalman împreună cu starea și observația sistemului.

Starea reținută de filtru sunt coordonatele 2D ale trăsăturii împreună cu viteza de deplasare pe cele două axe. Elementele de măsurat reprezintă poziția 2D. Modelul de actualizare al mișcării îl reprezintă legile mișcării, în care accelerația dintre două momente de timp se presupune a fi constantă. Ecuația prin care se actualizează starea filtrului ținând cont de starea precedentă, modelul zgomotului și eroarea în proces se găsește în (2). F este matricea de tranziție a stării, B este matricea modelului de control aplicat vectorului de control u_t , iar E_x modelează zgomotul prezent în proces. În (3) avem actualizarea observației pe baza stării estimate. H este matricea de tranziție a observației iar E_z reprezintă matricea de covarianță a unui model de zgomot gaussian alb de medie zero asociat observației. Valorile parametrilor sunt dați în Figura 8. Această aproximare nu reflectă realitatea, iar în cazul ideal ar trebui estimate din fiecare cadru imagine. Cu această configurație sunt folosite 3 filtre Kalman pentru estimarea poziției centrului palmei și poziției celor două puncte care definesc încheietura.

$$\begin{aligned} F &= \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad B = \begin{bmatrix} \frac{\Delta t^2}{2} \\ \Delta t \\ \Delta t \\ \Delta t \end{bmatrix} \\ H &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad E_x = \begin{bmatrix} \frac{\Delta t^4}{4} & 0 & \frac{\Delta t^3}{2} & 0 \\ 0 & \frac{\Delta t^4}{4} & 0 & \frac{\Delta t^3}{2} \\ \frac{\Delta t^3}{2} & 0 & \Delta t^2 & 0 \\ 0 & \frac{\Delta t^3}{2} & 0 & \Delta t^2 \end{bmatrix} \cdot \sigma_x \end{aligned}$$

Figura 8. Parametri filtrului Kalman folosiți de sistem.

Determinarea direcției de mișcare a mâinii se face folosind panta dreptei formate de locația centrului palmei din cadrul precedent și cel curent. Valoarea ei ne spune înclinația, iar semnul, dacă este crescătoare sau descrescătoare. Locațiile relative a celor două centre din cadre succesive ne spun dacă direcția este de la dreapta la stânga sau invers. În sistem, orice valoare aparținând $[1; \infty]$

sau $[-1; -\infty]$ se consideră a fi direcție verticală. O decizie finală se face prin vot majoritar asupra direcțiilor deduse din punctele consecutive ale traiectoriei mâinii. Determinarea momentului de început și sfârșit a traiectoriei mâinii, se face calculând viteza instantanee a mâinii între cadre succesive. Valoarea rezultată se compară cu un prag corespunzător vitezei minime, peste care mâna se află în mișcare. Mișcarea se sfârșește, când această valoare coboară sub pragul amintit.

Recunoașterea gesturilor dinamice

Un gest dinamic este alcătuit din direcția de deplasare a unui gest static, de-a lungul unui număr minim de cadre. Direcția, după felul în care se calculează, poate declanșa un eveniment către o interfață a aplicației client, însă componenta de recunoaștere statică detectează doar postura prezentă în cadrul curent. Se folosește un algoritm de monitorizare în timp a gesturilor pentru a da o decizie finală. El este aplicat pentru un gest static determinat și acceptă la intrare prezența, 1, sau lipsa lui, 0 în cadru. Algoritmii folosesc o coadă de lungime N care reține prezența sau absența lui în ultimele N cadre trecute. La fiecare cadru procesat, o nouă valoare de adevăr se adaugă la începutul listei și o valoare veche se elimină de la capătul ei. Când coada este plină se numără numărul maxim de valori de 1 consecutive din ea, n. Reținem, de asemenea, o variabilă Contor inițializată la 0. Dacă $n > Prag_{posibil}$ atunci $Contor = Temporizator_{posibil}$. Altfel dacă $n > Prag_{probabil}$ atunci $Contor = \max(Temporizator_{probabil}, n)$. În caz contrar se decrementează Contor. Dacă Contor este mai mare decât 0 atunci acel gest este considerat prezent în cadru, $Prag_{posibil} < Prag_{probabil} \leq N$.

Logica expusă se bazează pe continuitatea gesturilor în timp și pe o perioadă minimă de persistență. Astfel se resping cazuri de clasificări incorecte a unor gesturi și se tolerează situații în care un gest nu este prezent, dar avem informații trecute care indică prezența lui. Valorile alese pentru variabile sunt $N = 7$, $Prag_{posibil} = 3$, $Prag_{probabil} = 5$, $Temporizator_{posibil} = 5$, $Temporizator_{probabil} = 15$. În momentul detecției sfârșitului mișcării, se interoghează starea contorului și se returnează prezența sau absența gestului. Algoritmii sunt rulați individual pentru fiecare gest static modelat. Dacă sunt returnate mai multe gesturi atunci se alege acela cu variabila contor cea mai mare.

EVALUARE ȘI REZULATE

Evaluarea se face la nivelul: determinării încheieturii, recunoașterii dinamice a gesturilor.

Evaluarea algoritmului de determinare a încheieturii

Încheietura este cea care ne filtrează cea mai mare parte a vârfurilor degetelor false, însă aceasta se face prin creșterea timpului de procesare. O evaluare a algoritmului este necesară pentru a cuantifica acest lucru. Se folosește metodologia din [7] împreună cu baza de imagini, cea disponibilă la [15]. Conform ei pentru încheietura din tabela de adevăr |UV| și încheietura determinată de algoritm |U'V'| se calculează eroarea $e = |W'W| / |UV|$. W', W reprezintă mijloacele segmentelor. Se definesc cercurile $E=0.5$ și $E=1.0$, în care cifrele indică lungimea razei ca procentaj din segmentul |UV|. Se consideră că o

încheietură este corect găsită dacă eroarea e este mai mică decât constanta E aleasă. Procesul este ilustrat în Figura 10.

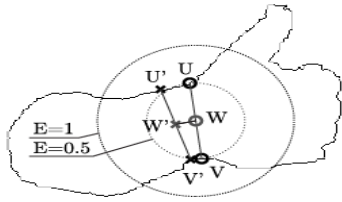


Figura 10. Măsurarea erorii. Se număra punctele din exteriorul primului cerc și cele din exteriorul celui de-al doilea[7]

Tabelul 2 prezintă rezultatele algoritmului nostru împreună cu cele publicate de autori în [7] pentru toată baza de date.

Tabel 1 Rezultatele evaluării detecției încheieturii

Denumire	LC[7]	Lucrare
Timp (ms)	32-81	22-52
$e > E, E=1.0$ (%)	14.7	12.11
$e > E, E=0.5$ (%)	36.0	23.4

Avantajul metodei din lucrare constă în determinarea implicită a orientării mâinii dată de dreptunghiul care înglobează silueta. Acest lucru face căutarea după încheietură mai ușoară. Timpul de procesare a fost măsurat pe un laptop având un procesor Intel Core 2 Duo T7300 și 2GB RAM și diferă de platforma utilizată de autori în [7].

Evaluarea gesturilor dinamice

În acest caz se utilizează baza de secvențe video folosită de lucrarea [9]. Un gest dinamic este alcătuit din direcția de deplasare a unui gest static, de-a lungul unui număr minim de cadre. Pentru a folosi baza de date, mișcările din secvențele video se definesc, ca o secvență de cele 4 direcții de mișcare, manual de către un operator uman. Pentru fiecare secvență video se verifică gesturile rezultate, cu cele definite pentru acea secvență. Eroarea de clasificare incorectă, reprezintă procentajul gesturilor dinamice care nu corespund clasificării manuale. Pentru fiecare secvență se măsoară timpul de procesare. Acești indicatori se calculează la nivelul întregii baze de date. Rezultatele sunt grupate după postura mâinii.

Tabel 2 Rezultate recunoaștere gesturi dinamice

Postură	Timp (ms)	Rata eroare (%)
Pumn	96-134	26
Palmă	89-127	13
Deget indicator	110-143	15

LIMITĂRI. ÎMBUNĂTĂȚIRI. CONCLUZII.

Limitările actuale ale sistemului se datorează algoritmilor folosiți. Segmentarea pe baza intervalului de culoare a pielii limitează fundalul în care se poate afla mâna. Lipsa clasificării între față și mână este un aspect negativ. Cazuri în care încheietura mâinii este puțin vizibilă și algoritmul sistemului dă greș, contribuie la rata de eroare. Mișcările reduse recunoscute sunt o altă limitare a sistemului.

În concluzie, lucrarea de față a prezentat o serie de soluții pentru a construi o aplicație capabilă să recunoască gesturi dinamice simple. Prin rezultatele obținute se arată că trăsăturile pot modela o varietate de gesturi, iar tehnicile de determinare a lor, permit o implementare eficientă, făcând posibilă înglobarea acestuia în sisteme mai mari de timp real.

REFERINȚE

1. Bishesh, K., Desire, S: Efficient kin detection under severe illumination changes and shadow, Intelligence Robotics and Applications, 2011.
2. Borgefors, G.: Distance transformations in digital images, Computer Vision, Graphics and Image Processing, 1986.
3. Chai, D., Ngan, K.N: Face segmentation using skin-color map in videophone applications, IEEE Transactions on Circuits and System for Video Technology, 1999.
4. Chaudhary, A., Raheja, J.L, Das, K.: Intelligent approaches to interact with machines using hand gesture recognition in natural way: a survey, International Journal of Computer Science and Engineering Survey, 2011.
5. Fakhreddine, K., Milad, A., Jamil, A.S., Mo, N.: Human-Computer Interaction: Overview on State of the Art, Canada International Journal on Smart Sensing and Intelligent Systems, 2008.
6. Ganapathyraju, S.: Hand gesture recognition using convexity hull defects to control an industrial robot, 3rd International Conference on Instrumentation Control and Automation, 2013.
7. Grzejszczak, T., Nalepa, J., Kawulok, M.: Real-time wrist localization in hand silhouettes, Conference on Computer Recognition System, 2013.
8. Kulshreshth, A., Zorn C., LaViola, J.J.: Poster: Real-time markerless kinect based finger tracking and hand gesture recognition for HCI, 3D IEEE Symposium on User Interfaces, 2013
9. Liu, L., Shao, L.: Learning Discriminative Representations from RGB-D Video Data, Proceedings International Joint Conference on Artificial intelligence, 2013
10. Maung, T.H.H.: Real-Time Hand Tracking and Gesture Recognition System using Neural Networks, Proceedings of World Academy of Science Engineering and Technology, 2009.
11. Mokhtar, M.H., Proamod, K.M.: Real time fingers and palm locating using dynamic circle templates, International Journal of Computer Applications, 2012.
12. Saxe, D., Foulds R.: Towards robust skin identification in video images, Proceedings 2nd conference of Automatic Face and Geture Recognition, 1996.
13. Welch, G., Bishop, G.: An introduction to the Kalman filter, Computer Science Technical Report, 1995
14. ***, Situl bibliotecii OpenCV, <http://opencv.willowgarage.com/wiki/>
15. ***, Sit către baza de imagini statice cu posturi ale mâinii, <http://sun.aei.polsl.pl/~mkawulok/gestures/>
16. ***, Algoritmul Brasenham pentru cerc, http://en.wikipedia.org/wiki/Midpoint_circle_algorithm