

Ovládanie počítača pohybom s využitím senzora

Leap Motion

Patrícia Szepesiová

Vedúci práce: RNDr. František Galčík, PhD., Konzultant: RNDr. Matej Nikorovič

Ústav informatiky, Prírodovedecká fakulta UPJŠ v Košiciach

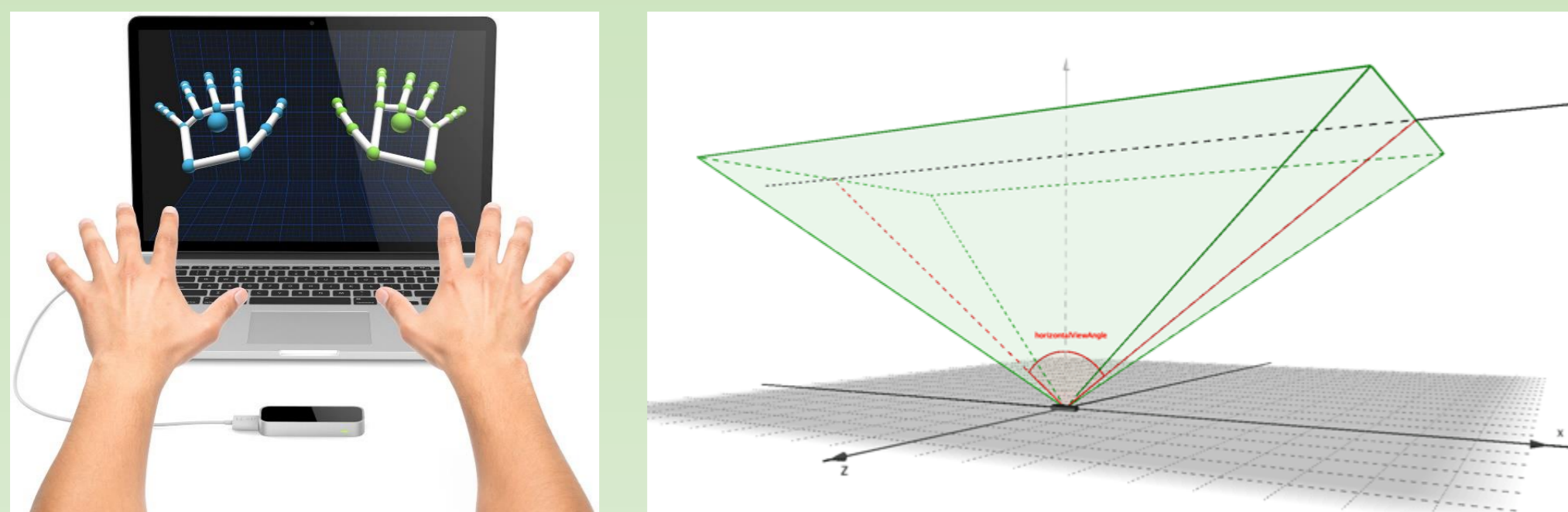
Abstrakt

Práca sa zaoberá návrhom a implementáciou nástroja na ovládanie počítača pohybom ruky s využitím senzora Leap Motion. Hlavným cieľom práce je implementovať nástroj, ktorý s dostatočnou presnosťou emuluje dotykové ovládanie. S využitím analytickej geometrie poskytuje ovládanie absolútneho pohybu kurzora, ktorý je závislý od vzájomnej polohy ruky a obrazovky počítača. Klasická, ale aj menej bežná funkcionálna zariadení na ovládanie počítača je nahradená gestami kurzora, na ktoré je aplikované rozpoznávanie.

Úvod

Bezdotykové ovládanie počítača v porovnaní s dotykovým rieši otázku hygieny a bezpečnosti, pretože odstraňuje potrebu dotyku s dotykovou plochou zariadenia, ktorá predstavuje zdroj nákaz a ochorení. Obzvlášť rizikové sú verejné obrazovky, ktorých sa denne dotýka nespočetne množstvo ľudí. Práca sa skladá z troch hlavných celkov, a to kalibrácia polohy obrazovky, výpočet pozície kurzora na obrazovke a rozpoznávanie gest kurzora.

Na zaznamenanie polohy a pohybu rúk využívame zariadenie Leap Motion [1], ktoré sníma ruky pomocou dvoch infračervených kamier. Snímky následne prevádza matematickými algoritmi na 3D súradnice kĺbov rúk, ktoré sa nachádzajú v zornom poli senzora (obrázky 1 a 2). Priemerná presnosť je 0,7 mm podľa štúdie z roku 2013 [3].

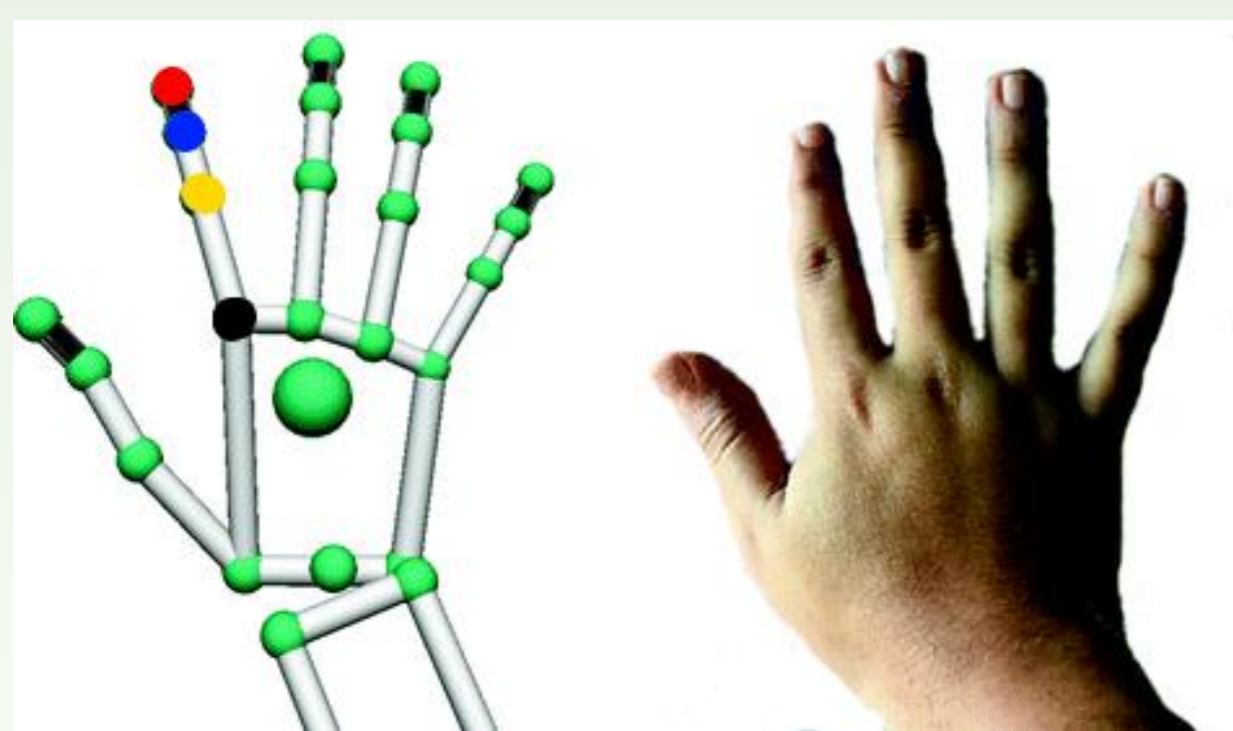


Obrázok 1: Model ruky v zornom poli senzora. Obrázok 2: Zorné pole senzora Leap Motion.

Kalibrácia

Prvým krokom pred začatím používania nástroja na ovládanie počítača bez dotyku je získať polohu obrazovky v zornom poli senzora Leap Motion. Plochu popisujeme parametricky pomocou troch bodov. Zvolili sme tri body, ktoré sa nachádzajú v čo najväčšej vzdialenosti – súradnice troch rohov obrazovky, aby sme minimalizovali odchýlky spôsobené nepresnosťami pri získavaní 3D súradníc týchto bodov.

Súradnice každého z troch bodov získavame tak, že 5-krát ukážeme na daný bod, vždy pod iným uhlom. Pri každom ukázaní zaznamenáme súradnice dvoch kĺbov ukazujúceho prsta (na obrázku 2 zvýraznené červenou a žltou farbou) a preložíme nimi priamku.



Obrázok 3: Kĺby ruky v zornom poli senzora.

V ideálnom prípade by sa tieto priamky pretli v bode, ktorého súradnice hľadáme. Kvôli nepresnostiam a ľudskému faktoru sa priamky takmer určite nepretnú. Namiesto prieniku hľadáme bod, ktorého vzdialenosť je od všetkých priamok najmenšia možná.

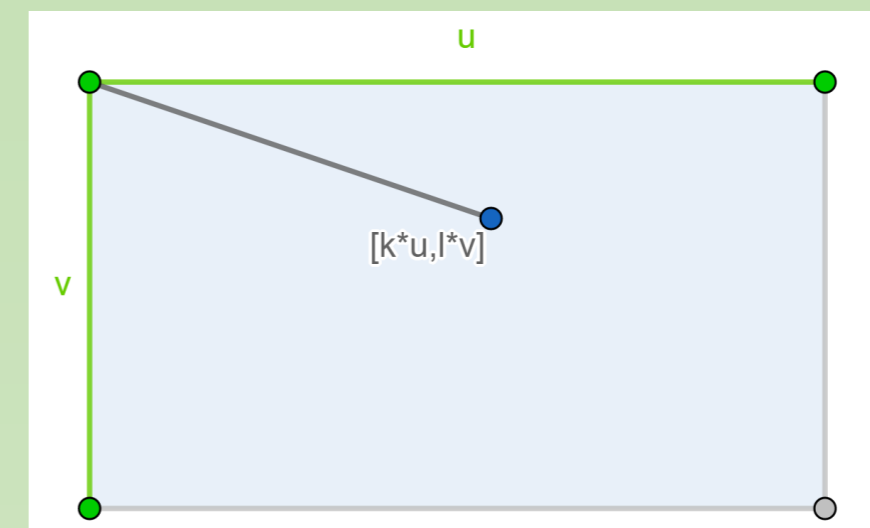
Na získanie 3D súradníc daných bodov využívame štatistickú metódu regresnú analýzu. Metóda hľadá minimum globálnej chyby, ktorou je súčet štvorcov vzdialeností hľadaného bodu od každej z priamok. Majme n priamok, pričom i -ta priamka je vyjadrená bodom $[a_{i1}, a_{i2}, a_{i3}]$ a vektorom (v_{i1}, v_{i2}, v_{i3}) . Súradnice $[x_1, x_2, x_3]$ hľadaného bodu získame vyriešením nasledujúcej maticovej rovnice:

$$\begin{pmatrix} \sum_{i=1}^n \frac{2v_{i1}^2 + 2v_{i2}^2 + 2v_{i3}^2}{v_{i1}^2 + v_{i2}^2 + v_{i3}^2} & \sum_{i=1}^n \frac{-2v_{i1}v_{i2}}{v_{i1}^2 + v_{i2}^2 + v_{i3}^2} & \sum_{i=1}^n \frac{-2v_{i1}v_{i3}}{v_{i1}^2 + v_{i2}^2 + v_{i3}^2} \\ \sum_{i=1}^n \frac{-2v_{i1}v_{i2}}{v_{i1}^2 + v_{i2}^2 + v_{i3}^2} & \sum_{i=1}^n \frac{2v_{i1}^2 + 2v_{i2}^2 + 2v_{i3}^2}{v_{i1}^2 + v_{i2}^2 + v_{i3}^2} & \sum_{i=1}^n \frac{-2v_{i2}v_{i3}}{v_{i1}^2 + v_{i2}^2 + v_{i3}^2} \\ \sum_{i=1}^n \frac{-2v_{i1}v_{i3}}{v_{i1}^2 + v_{i2}^2 + v_{i3}^2} & \sum_{i=1}^n \frac{-2v_{i2}v_{i3}}{v_{i1}^2 + v_{i2}^2 + v_{i3}^2} & \sum_{i=1}^n \frac{2v_{i1}^2 + 2v_{i2}^2 + 2v_{i3}^2}{v_{i1}^2 + v_{i2}^2 + v_{i3}^2} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^n \frac{2a_{i1}(v_{i2}^2 + v_{i3}^2) - 2v_{i1}(a_{i2}v_{i2} + a_{i3}v_{i3})}{v_{i1}^2 + v_{i2}^2 + v_{i3}^2} \\ \sum_{i=1}^n \frac{2a_{i2}(v_{i1}^2 + v_{i3}^2) - 2v_{i2}(a_{i1}v_{i1} + a_{i3}v_{i3})}{v_{i1}^2 + v_{i2}^2 + v_{i3}^2} \\ \sum_{i=1}^n \frac{2a_{i3}(v_{i1}^2 + v_{i2}^2) - 2v_{i3}(a_{i1}v_{i1} + a_{i2}v_{i2})}{v_{i1}^2 + v_{i2}^2 + v_{i3}^2} \end{pmatrix}$$

Výpočet pozície kurzora

Výpočet súradníc kurzora na obrazovke pozostáva z dvoch krokov. V prvom kroku počítame prienik priamky, ktorú získavame preložením cez dva body predstavujúce kĺby ukazujúceho prsta, s plochou obrazovky, ktorej polohu sme získali kalibráciou. Získavame 3D súradnice kurzora.

V druhom kroku prevádzame 3D súradnice do 2D súradníc kurzora na obrazovke jednoduchým zobrazením. Plochu obrazovky máme popísanú troma bodmi, ktorých 3D súradnice sme získali kalibráciou. Poznáme však aj ich 2D súradnice na obrazovke, získali sme ich z rozlíšenia monitora. Na obrázku je vidieť vyjadrenie pozície kurzora pomocou vektorov u, v a parametrov k, l . Parametre k a l sú hodnoty z intervalu $\langle 0, 1 \rangle$, aby prienik s rovinou patrilo ploche obrazovky. Pre daný bod prieniku budú parametre k a l pre 2D a 3D vyjadrenie rovnaké, preto ich získame z 3D prieniku a dosadením do 2D vyjadrenia získame pozíciu kurzora na obrazovke.



Obrázok 4: Vyjadrenie pozície kurzora na obrazovke pomocou vektorov.

Rozpoznávanie gest

Okrem ovládania kurzora, ktorému sme sa doposiaľ venovali, je potrebné implementovať funkcionálnu, ktorá nahrádza klikanie myši, posúvanie obsahu, alebo rôzne klávesové skratky. Na posúvanie obsahu nahor a nadol využívame gesto, ktoré je už implementované v knižnici pre Leap Motion, a to krúženie prstom v smere a proti smeru hodinových ručičiek. Rovnako využívame už implementované gesto rýchleho pohybu rukou vpravo alebo vľavo, ktoré emuluje stlačenie šípok vpravo alebo vľavo. Po testovaní rôznych možností pre emuláciu kliknutia, ktorá musí byť pre používateľa čo najprirodzenejšia, sme sa rozhodli pre emuláciu klikania na pomyselnú dotykovú plochu, ktorá sa nachádza niekoľko centimetrov pred reálnou plochou obrazovky. Teda kliknutie nastáva, keď sa vzdialenosť ukazujúceho prsta od obrazovky dostane pod určitú vzdialenosť.

Ďalšiu funkcionálnu sme nahradili gestami kurzora. Gestom kurzora rozumieme sekvenciu smerov, v ktorých sa kurzor v čase pohybuje. Rozpoznávanie gest je klasifikačná úloha, ktorá pozostáva z dvoch fáz – fáza učenia a fáza klasifikácie.

Rozhodli sme sa využiť jednoduchý klasifikátor k -najbližších susedov. Spočíva v porovnaní vykonaného gesta s tréningovou množinou. Vyberieme k vzoriek, s ktorými sa najviac zhoduje a ak nadpolovičná množina z nich je priradená jednému gestu, prehlásime, že sa dané gesto rozpoznalo.

Vo fáze učenia vykonávame a ukladáme každé gesto 5-krát. Na to, aby sme gesto vedeli uložiť a následne porovnať, musíme vypočítať jeho vektor príznakov. Tým je sekvencia smerov, ktorými sa pri vykonávaní kurzor pohybuje v maximálnej dĺžke 3-5 sekúnd. V prvom rade je potrebné začiatok a koniec vykonávania gesta ohraničiť, a to krátkym sekundovým ukazovaním prsta na jedno miesto. V ich rozmedzí zbierame pozície kurzora rýchlosťou približne 20 za sekundu. Následne aplikovaním Douglas-Peucker algoritmu [2] zjednodušíme zoznam zozbieraných pozícií a tiež vyhladáme drobné odchýlky, vzniknuté zmenou smeru kurzora na veľmi malom úseku.

Následne pre každú dvojicu za sebou idúcich pozícií určíme jeden z 8 smerov (nahor, nadol, vpravo, vľavo, vľavo-hore, vpravo-hore, vľavo-dole, vpravo-dole). Pomocou smernice priamky, ktorá danými bodmi prechádza a ich x -ových a y -ových súradníc vieme jednoznačne povedať, ku ktorému z 8 smerov má vypočítaný smer najbližšie.

Podobnosť vektora príznakov vykonaného gesta s uloženými vzorkami vo fáze klasifikácie určíme ako Levenshteinovú vzdialenosť [4], tiež známu ako algoritmus *edit distance*. Po jej vypočítaní pre každú vzorku z tréningovej množiny vyberáme 5 takých, ktorých vzdialenosť je najmenšia a zároveň neprekračuje povolenú hodnotu. Na základe toho povieme, či bolo nejaké gesto rozpoznané, alebo nie. Úspešne sa nám podarilo rozpoznávať základné útvary, ako sú štvorec, či kruh.

Záver

Zaoberali sme sa výpočtom pozície kurzora na obrazovke z polohy ruky v zornom poli senzora, čo nám umožňuje jeho ovládanie. Pri precíznej kalibrácii dosahujeme uspokojivú presnosť za predpokladu, že ukazujúca ruka je v takej polohe, v ktorej vie senzor s dostatočnou presnosťou určiť pozície jej kĺbov. Implementovali sme bežnú funkcionálnu ovládania počítača, akou je klikanie a posúvanie obsahu. Venovali sme sa rozpoznávaniu gest, ktoré funkcionálnu ovládania rozširuje.

Literatúra

- [1] Leap Motion, Inc. Leap motion controller. <https://www.leapmotion.com>, 2015.
- [2] DOUGLAS, David H.; PEUCKER, Thomas K. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 1973, 10.2: 112-122.
- [3] WEICHERT, Frank, et al. Analysis of the accuracy and robustness of the leap motion controller. *Sensors*, 2013, 13.5: 6380-6393.
- [4] LEVENSHTEIN, Vladimir I. Binary codes capable of correcting deletions, insertions, and reversals. In: *Soviet physics doklady*. 1966. p. 707-710.

Zdroje obrázkov:

https://media.springernature.com/original/springer-static/image/chp/%3A10.1007%2F978-3-319-26410-3_22/MediaObjects/385589_1_En_22_Fig1_HTML.gif

<https://3bonlp1aiidtbao4s10xacvn-wpengine.netdna-ssl.com/wp-content/uploads/2017/03/leap-motion-3d-motion-gesture-controller-10-large.jpg>

https://di4564baj7skl.cloudfront.net/documentation/images/Leap_horizontalViewAngle.png