

Univerza v Ljubljani
Fakulteta *za elektrotehniko*



Merjenje stresa preko računalniške periferije

Seminar pri predmetu Avtomatizirani in virtualni merilni sistemi

Adam Hrastnik, 64150251

LJUBLJANA, MAJ 2020

Vsebina

1	Uvod	2
2	Simulacija različnih razmer	3
2.1	Simulacija stresnih razmer.....	3
2.2	Simulacija ne-stresnih razmer.....	3
3	Matematični model in algoritem sistema.....	4
3.1	Delovanje modela in algoritma.....	4
3.2	Prilagajanje ocene stresa glede na uporabnika in aplikacijo	5
4	Uporabniški vmesnik.....	6
4.1	Zasnova zajema interakcije.....	6
4.2	Zajem gibanja računalniške miške	8
5	Rezultati.....	8
5.1	Analiza v frekvenčnem prostoru	8
5.2	Odziv v stresnih razmerah	10
5.3	Odziv v ne-stresnih razmerah	10
6	Zaključek	12

Povzetek

Prekomeren stres lahko v našem življenju povzroči množico težav – v obliki zdravstvenih ali socialnih. Projekt se osredotoča predvsem na zaznavanje stresnih situacij, za nastanek katerih je razlog programska oprema. Merjenje stresa se izvaja s pomočjo standardnih računalniških perifernih enot – specifično, miške in tipkovnice. Predstavljen je matematični model, ki oceni nivo stresa iz količine gibanja miške. Prikazana je tudi izvedba zajema interakcije z računalniško periferijo. Na koncu je uporaba modela ovrednotena s pomočjo računalniških iger, na katerih so opisane omejitve modela.

Ključne besede: stres, model, računalniška periferija, programska oprema, računalniške igre

Abstract

Excessive stress can cause a plethora of problems in our lives – in the form of health or social issues. This project mainly focuses on measuring stressful situations caused by computer software. Stress measurement is carried out, by utilizing common computer peripherals – more specifically a computer mouse and keyboard. A mathematical model is presented, which estimates the level of stress from the quantity of mouse motion. Also, the development of a recorder for computer peripheral interaction is displayed. At the end, the model is verified with the use of computer games, upon which the model limitations are described.

Key words: stress, model, computer peripherals, software, computer games

1 Uvod

Stres je del našega življenja, kateremu se težko povsem izognemo. Prekomeren stres lahko pripelje do zdravstvenih težav [6] [7] [8] [9], zato je pomembno, da je prisotnost tega v našem življenju, kar se da minimalna. Povečan stres nas lahko doleti zaradi mnogih različnih faktorjev v našem življenju. V tem projektu je v ospredju stres, ki je povezan z uporabo osebnega računalnika, oz. bolj specifično – programske opreme. Če vzamemo za primer razne uporabniške vmesnike, je tam ključnega pomena enostavnost uporabe, ki nove uporabnike ne odvrne od aplikacije, izkušenim pa omogoča bolj učinkovito uporabo. Slabo načrtani vmesniki lahko pripeljejo do nezadovoljstva uporabnikov in tudi stresa [12]. Uporaba modela za oceno stresa, bi tako pripomogla k razvoju aplikacij in s tem uspešnosti podjetji, ki stojijo za njimi.

Precej rešitev že obstaja za ne-invazivno merjenje stresa, s pomočjo posebno prirejenih perifernih enot [1] [2] [3] [4]. Težava pri teh je, da potrebuje vsak uporabnik kupiti posebno prirejen izdelek. Za manjše, specifične aplikaciji, to navadno ni problem, toda če poskušamo prodati rešitev veliki množici uporabnikov, je velika verjetnost, da jih večina tovrstne rešitve ne bo sprejela.

Čeprav ostaja to področje dokaj neraziskano, so se raziskovalci na univerzi Berkeley v Kaliforniji te težave zavedali, zato so podobno rešitev kot je obravnavana, že razvili s pomočjo mehansko impedančnega modela »mass-spring-damper« [5]. Pomanjkljivost, ki je v njihovi študiji je, da so model preizkušali na zelo preprostih in generičnih preizkusih – oz. v naprej pripravljenih laboratorijskih preizkusih. Za razliko od omenjene raziskave, poskuša ta projekt razviti model, ki bo deloval tudi v realnih aplikacijah.

Projekt temelji na predpostavki, da povečan stres vpliva na način uporabe računalniške periferije. Če to drži, je mogoče oceniti nivo stresa, zgolj iz analize uporabe standardne računalniške periferije, kot sta miška in tipkovnica. Pri čemer se projekt posveti predvsem analizi gibanja računalniške miške.

2 Simulacija različnih razmer

Zanesljivo in ponovljivo vzbujanje stresa z namenskimi stresorji, je dokaj težavna naloga [5]. Za dober psihološki stresor so se izkazale računalniške igre [13], ki so s pridom izkoriščene v tem projektu.

2.1 Simulacija stresnih razmer

Kot stresor je izbrana računalniška igra »Slender man«, katere glavni učinek je strah. Cilj igre je zbrati 8 listov papirja v temačnem gozdu, med tem ko se postopoma bolj pogosto pojavlja in približuje lik »Slender man«. Če se ta igralcu preveč približa, je igre konec. Med posameznimi pojavi, je igra dokaj umirjena, razen atmosfere, katere napetost se prav tako stopnjuje. Napetost tako eksponentno narašča z vsaki zbranim listom, kar je tudi razlog za izbiro te igre. Namreč, podobno kot napetost postopoma narašča stres, zato je proti koncu ta lahko dokaj znaten.

Ker je igra dokaj kratka, sem za simulacijo stresnih razmer odigral celotno igro, ki je trajala dobrih 7 min. Med igranjem so bili štirje bolj stresni trenutki, ob nenadnih pojavih lika »Slender man«.

2.2 Simulacija ne-stresnih razmer

Polarno nasprotje stresorja, je igra »Minetest«, katera ne vsebuje nobene namenske napetosti. Igralec je spuščen v svet narave, kjer si določi cilj sam. Glavni namen igre je, da si igralec sprostí domišljijo in zgradi kar si zaželi, s pomočjo materialov, ki jih ima na voljo okoli sebe.

Kot simulacijo ne-stresnih razmer sem v tej igri zgradil manjšo hišo, za katero sem pred tem nabral vse potrebne materiale. Celotna simulacija je trajala slabih 10 min. Medtem nisem naletel na praktično nobeno stresno situacijo, razen na manjšo pri uporabi kamere, katere občutljivost je bila previsoka in je ovirala gradnjo.

3 Matematični model in algoritem sistema

Z ustrezno načrtanim matematičnim modelom, je mogoče predvideti stanje sistema, glede na podane vhode. Pri uporabi kompleksnih modelov, ki potekajo v več stopnjah in se izvajajo na računalniku, se te zapiše v obliki algoritmov.

3.1 Delovanje modela in algoritma

Algoritem neprestano zbira podatke o interakciji s periferijo. Podatke obdeluje po posameznih oknih, oz. odsekoma, kar je pomembno pri zaznavanju spremembe o interakciji. Okno se neprestano pomika z vsakim zajetim vzorcem.

Matematični model deluje na principu modificiranega izračuna statistične z-vrednosti (1). Vzorec f_{sw} predstavlja povprečna minimalna vzorčna frekvenca v trenutnem oknu, ki je dolžine 2 s. Tu je izkoriščen poseben način, ki ga operacijski sistem uporabi za zajem pozicije miške, ki je bolj podrobno opisan v kasnejšem poglavju (4.2). S tem načinom je mogoče pridobiti količino gibanja samo iz časovnega razmaka med posameznimi vzorci.

$$z_{sw} = \frac{f_{sw} - \overline{f_{sw}}}{\sigma_{sw}} - \sigma_{sw} \quad (1)$$

V splošnem faktor stresa s_f (2) predstavlja normalizirano količino gibanja, z izluščenim odstopanjem od pozitivnega standardnega odklona σ_{sw} (1).

$$s_f = \begin{cases} 0; & z_{sw} \leq 0 \\ z_{sw}; & z_{sw} > 0 \end{cases} \quad (2)$$

Dodatno, se je potrebno zavedati analogne narave stresa. Namreč, stres, po tem ko naraste – ne upade takoj, ko se uspemo izogniti situaciji, ki ga je povzročila. Za namen simulacije upada stresa, se uporabi eksponentno utežno povprečje (3), ki postopoma niža vrednost ocenjenega faktorja stresa v časovnem območju ~ 15 s, medtem ko takoj aplicira vse vrednosti stresa, ki so večje od prejšnje.

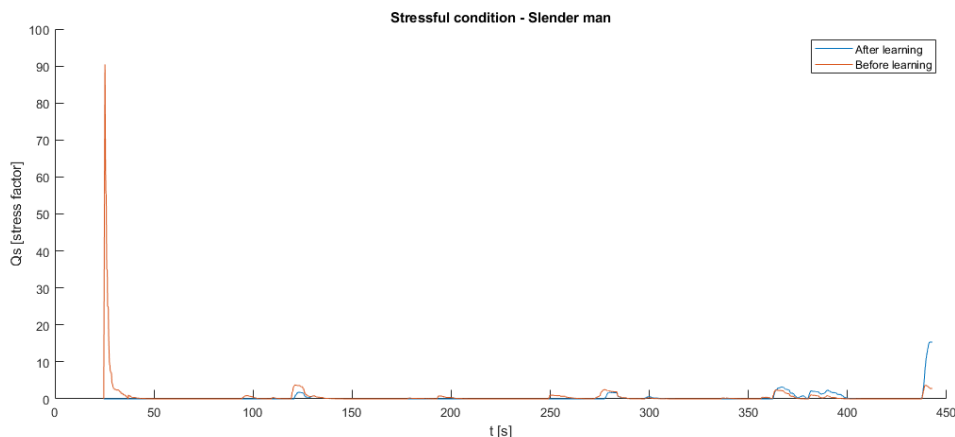
$$\sigma_s^n = \begin{cases} s_f; & s_f > \sigma_s^{n-1} \\ s_f \cdot w + \sigma_s^{n-1} \cdot (1 - w); & s_f \leq \sigma_s^{n-1} \end{cases} \quad (3)$$

3.2 Prilagajanje ocene stresa glede na uporabnika in aplikacijo

Sistem deluje v dveh načinih, in sicer v načinu *učenja* in *izvajanja*. V obeh načinih se operira nad parametri povprečne vzorčne frekvence in standardnega odklona (1). Podobno kot pri izračunu faktorja stresa (3), se tudi tukaj uporabi eksponentno utežno povprečje.

Učenje se izvaja, ko se algoritmu ne poda vhodnih parametrov – povprečna vzorčna frekvenca in standardni odklon. V tem načinu se ti dve vrednosti računata na že pridobljenih podatkih, kar pomeni, da izračunan stres v začetku uporabe, ne odseva realne vrednosti – vse dokler se vrednosti ne ustalita.

Načina izvajanja se razlikujeta samo v vrednosti uteži w za izračun utežnega povprečja. Pri učenju je vrednost uteži dokaj višja, kar pomeni, da ima nov vzorec dokaj večji vpliv na končni rezultat.



Slika 1: Primerjava ocenjenega faktorja stresa pred in po učenju (Slender man)

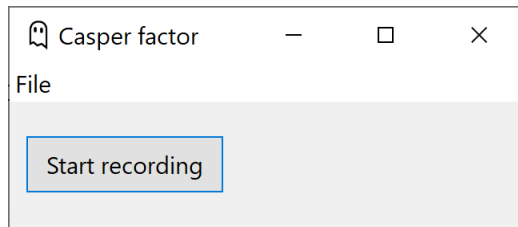
V simulaciji je učenje izvedeno, tako da je na istem odzivu algoritem pognan dvakrat. Prvič brez začetnih parametrov povprečnega gibanja in standardnega odklona, drugič pa z vrednostmi, ki jih je vrnila prva simulacija.

Če primerjamo razliko pred in po učenju, je očitno, da učenje konkretno izboljša končno oceno faktorja stresa.

4 Uporabniški vmesnik

Glavni namen uporabniškega vmesnika je zajem in izvoz interakcije s periferijo. Zajeto interakcijo s periferijo, je mogoče izvoziti v datoteko, ki je združljiva s programom Matlab. Prvotno je bil namen dodati uporabniškemu vmesniku tudi prikaz ocene stresa v realnem času, toda zaradi časovne omejenosti to ni uspelo.

Vmesnik je zasnovan v programskem jeziku Python. Razlog za izbrani programski jezik je, da omogoča z njegovo preprosto sintakso hitro zasnovano raznovrstnih aplikacij in vmesnikov. Sicer Python omogoča celotno analizo in razvoj vmesnika, toda je analiza podatkov še vedno bolj priročna in hitrejša v okolju Matlab.



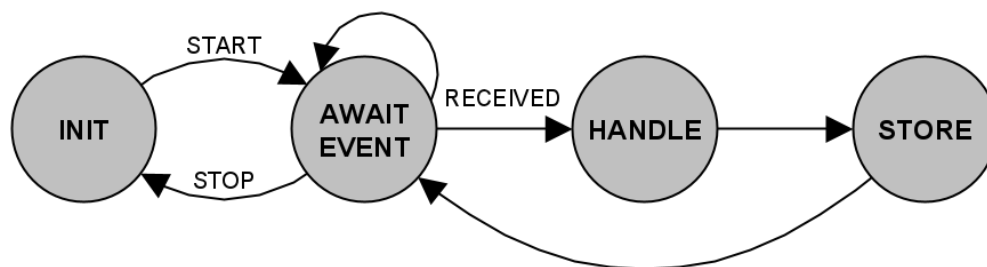
Slika 2: Uporabniški vmesnik

4.1 Zasnova zajema interakcije

Preden lahko analiziramo interakcijo s periferijo, jo potrebujemo najprej zajeti. Za Python so na voljo številne različne knjižnice, med katerimi so tudi knjižnice za neposredno interakcijo s periferijo. V tem projektu je uporabljena knjižnica *pynput* [10]. Izbrana je bila zaradi njene edinstvene nizko-nivojske povezave z operacijskim sistemom – kar omogoča zajemanje z minimalnimi zakasnitvami.

Zajemalnik podpira zajemanje interakcije z miško in tipkovnico. To pomeni, da si shranjuje vse pritiske gumbov na tipkovnici in miški ter gibanje miške. Operacijski sistem ob vsaki interakciji s periferijo proži dogodek. Pri tipkah se sproži dogodek, tako ob pritisku tipke, kot pri spustu. Pri gibanju miške, pa je postopek rahlo bolj kompleksen, saj sistem poveča proženje dogodkov pri kompleksnejših gibanjih. Torej, če z miško vlečemo ravne črte sistem proži manj dogodkov, kot če vlečemo krivulje.

Zajem interakcije deluje kot popolnoma ločen proces od ostalega vmesnika. Proces zajema komunicira z vmesnikom preko sinhronizirane vrste [11], kar omogoča nemoteno delovanje vmesnika, kljub nastalim zakasnitvam zaradi zajema.



Slika 3: Delovanje zajemalnika interakcije

Ob zagonu aplikacije, se zajemalnik nahaja v stanju »INIT« (Slika 1), kjer se vzpostavijo vse podatkovne zbirke in začetna stanja. Ko uporabnik pritisne gumb za začetek zajemanja, se zažene zajemalnik ter s tem preide v stanje »AWAIT EVENT«, kjer čaka na proženje perifernega dogodka. Ob proženju, preide v stanje »HANDLE«, kjer se dogodek ustrezno obdela, glede na njegov tip. Različni dogodki podajo drugačne informacije, od katerih nekatere potrebujejo biti pretvorjene. Po obdelavi, se podatki shranijo v stanju »STORE«, zatem pa se zajemalnik povrne v stanje »AWAIT EVENT«, kjer čaka na naslednji dogodek. V primeru, da uporabnik ustavi zajemanje, se zajemalnik ustavi tako, da se vrne v prvotno stanje in ponastavi vse zbirke in notranja stanja.

4.2 Zajem gibanja računalniške miške

Sistem vrne koordinate kazalca miške v enoti slikovnih elementov. Če želimo analizirati gibanje miške, nastane težava, ker imajo različni zasloni različne resolucije in velikosti. Zaradi tega razloga je potrebno koordinate poenotiti na tak način, da bo gibanje neodvisno od lastnosti zaslona. Poenotenje je možno izvesti s pretvorbo koordinat iz enote slikovnih točk v milimetre. K sreči zasloni že vsebujejo podatke o velikosti in resoluciji, zato postane pretvorba koordinat trivialna, z uporabo preprostega križnega računa (4).

$$\overrightarrow{p_{mm}} = \frac{\overrightarrow{p_{pixel}} \cdot \overrightarrow{s_{mm}}}{s_{pixel}} \quad (4)$$

Posebno pozornost je potrebno posvetiti tudi vzorčni frekvenci miške. Navadno imajo miške vzorčne frekvence, ki se gibljejo od okrog 100 Hz do 1 kHz. Nekateri bolj posebni modeli, lahko uporabljajo tudi višje frekvence, toda te se navadno uporabljajo samo v laboratorijskih aplikacijah pri psihofizioloških raziskavah. V tem primeru se uporablja vzorčna frekvenca 1 kHz.

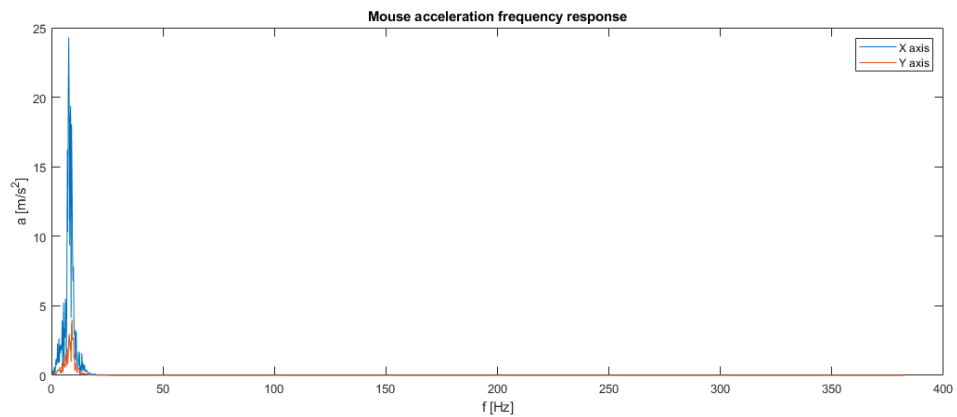
5 Rezultati

Prvotna ideja je bila oceniti nivo stresa, s pomočjo Fourierve transformacije pospeškov gibanja miške, toda rezultat žal ni prikazal pričakovanih vrednosti. Po analizi vmesnih rezultatov, se je že sama povprečna količina gibanja miške v posameznem oknu izkazala za dober indikator stresa – oz. vsaj v obravnavani simulaciji stresa.

Postopki in opisi simulacij so opisani v prejšnjem poglavju (2).

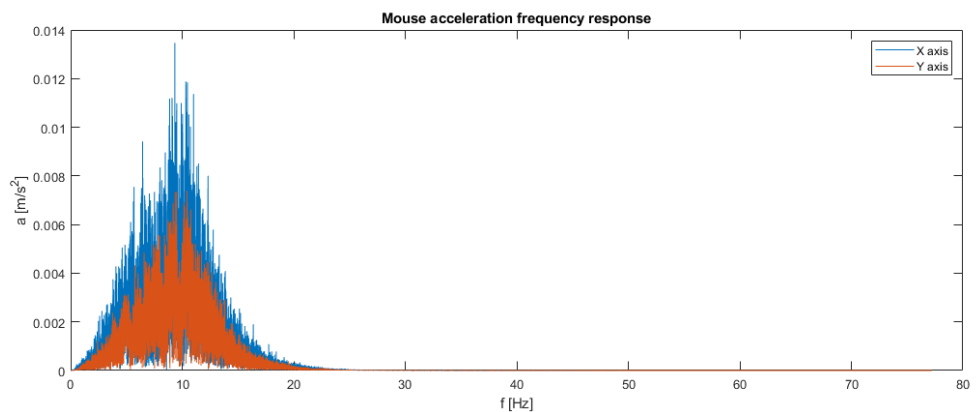
5.1 Analiza v frekvenčnem prostoru

Čeprav je bilo mogoče zaznati periodičnost gibanja v razmerah, ki so pretežno vsebovale periodično gibanje, se je v realnih razmerah periodičnost skoraj popolnoma izgubila med šumom.



Slika 4: Frekvenčni odziv gibanja miške pri namenskem visoko-frekvenčnem gibanju

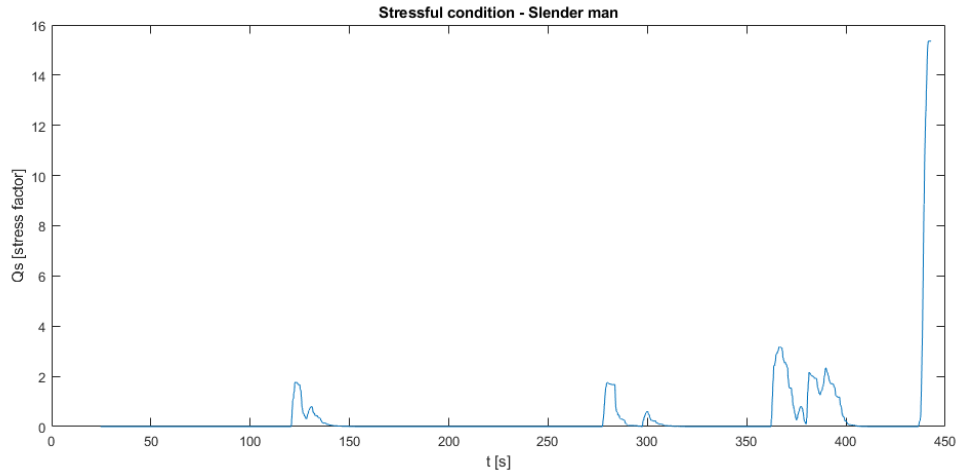
Pri namenskem gibanju miške z visoko frekvenco, je mogoče opaziti, da je razvidna frekvenca $f \approx 7 \text{ Hz}$, pri kateri je bilo gibanje izvedeno.



Slika 5: Frekvenčni odziv v realnih razmerah

Rezultati so precej različni v realnih razmerah, kjer je graf tako nasičen, da iz njega niso razvidni posebni dogodki. Težava je delno v diskretizaciji pozicije miške, toda ta je minimalna, saj je pozicija močno filtrirana. V glavnem na podatke vpliva dejstvo, da v daljšem časovnem obdobju izvedemo veliko različnih gibanj, iz katerih je težko določiti periodičnost. Preizkusiti bi bilo potrebno še z uporabo oknenja, kot pri predlaganem algoritmu.

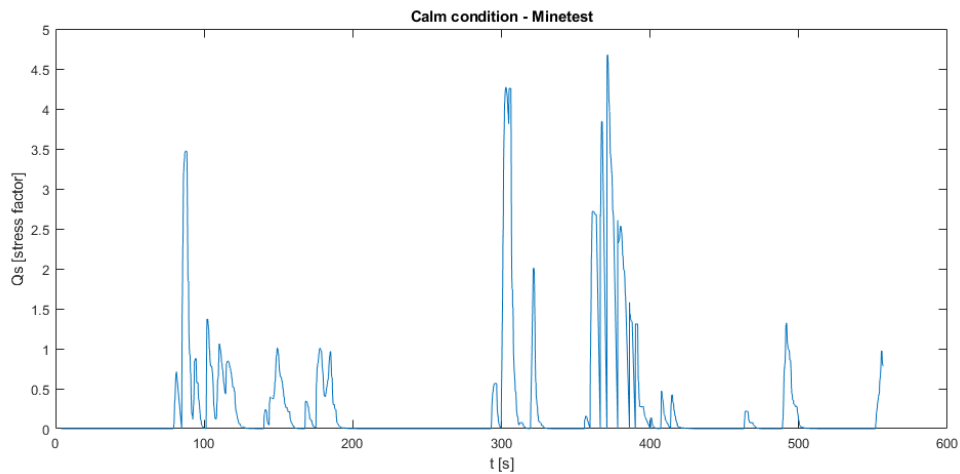
5.2 Odziv v stresnih razmerah



Slika 6: Faktor stresa v stresni situaciji

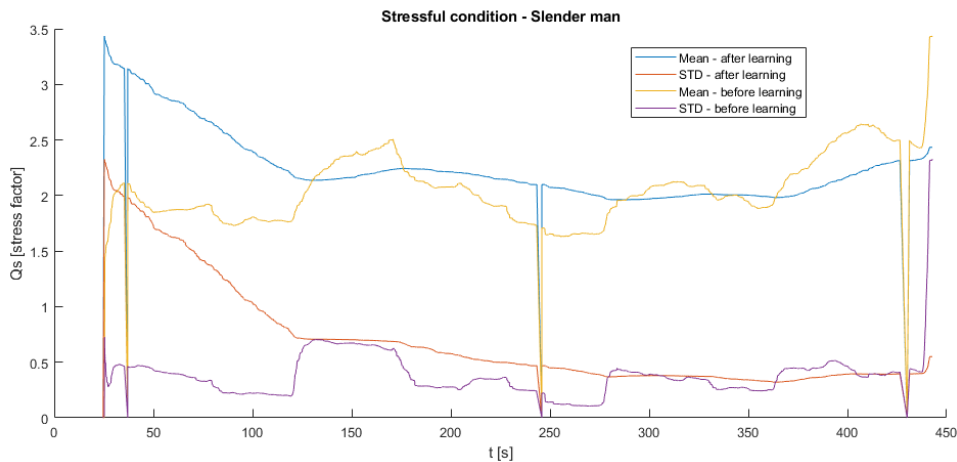
Tako za simulacijo stresnih razmer, kot ne-stresnih, je za zagotovitev objektivne analize zajet posnetek interakcije. Iz posnetka in ocenjenega faktorja stresa, lahko razberemo, da v tej aplikaciji model zelo dobro odraža realni nivo stresa.

5.3 Odziv v ne-stresnih razmerah



Slika 7: Faktor stresa v ne-stresni situaciji

Iz grafa je razvidno, da so nihanja ocenjenega stresa pogosta in relativno velika po amplitudi, kar definitivno ne odseva realne vrednosti stresa, glede na posnetek interakcije. V tem primeru bi moral ocenjen stres le rahlo nihati (oz. sploh ne) okrog abscisne osi. Model torej v tem primeru ne ustreza.



Slika 8: Primerjava med povprečno vzorčno frekvenco in standardnim odklonom pred in po učenju

Iz primerjave je razvidna ustalitev povprečne vzorčne frekvence in standardnega odklona, po učenju – kar je rezultat drugačne vrednosti uteži za eksponentno utežno povprečje.

Vmesne točke, katerih vrednost je nič, se pojavijo, kadar v posameznem oknu ni bilo zaznanega gibanja.

6 Zaključek

Iz rezultatov je mogoče opaziti, da se v primerih izbrane stresne in ne-stresne situacije, uspešnost ocene stresa znatno razlikuje. To se zgodi, kljub temu, da sta obe igri izbrani skrbno, saj se njuna načina igranja v splošnem ne razlikujeta. Torej, nobena od iger ni intenzivna in obe se igrata na podoben način. Glavna razlika je, da ne-stresna igra zahteva večjo mero interakcije z igralcem, saj vsebuje mehanike, ki že same po sebi zahtevajo večjo količino gibanja miške. Sklenemo lahko, da sistem najbolje deluje v primerih, ko količina gibanja v stresnih situacijah znatno odstopa od povprečne količine gibanja – medtem ko ta ostaja dokaj konstantna in nizka.

Z nadaljevanjem projekta, bi bilo ključnega pomena model razširiti tako, da bi poleg količine upošteval tudi način gibanja miške in iz tega poskušal oceniti nivo stresa. Tovrstna implementacija bi lahko pripomogla, k uspešnosti modela v različnih aplikacijah. Pomemben korak bi bila tudi podrobnejša validacija modela. Namreč, model sem preizkušal samo na samem sebi, iz česar ni mogoče skleniti popolnoma objektivnih in ponovljivih rezultatov; zato bi bilo potrebno model preizkusiti na množici ljudi. Da pa bi sploh lahko model potrdili kot objektivno uspešen, bi bilo tega potrebno primerjati z eno od že preverjenih in zanesljivih metod [14].

Literatura

- [1] Jaka Ogorevc, Tomaž Lavrenčič, Dragan Pršlja, Gregor Geršak, Valentin Batagelj, Janko Drnovšek, Univerza Ljubljani, Fakulteta za elektrotehniko, 2012
- [2] Gregor Geršak, Enostavni nizkocenovni merilniki prevodnosti kože, Univerza Ljubljani, Fakulteta za elektrotehniko, 2013
- [3] T. Lavrenčič, Švicmiš - sistem za merjenje prevodnosti kože, diplomska naloga, Univerza Ljubljani, Fakulteta za elektrotehniko, 2012
- [4] Javier Hernandez, Pablo Paredes, Asta Roseway, Mary Czerwinski, Under Pressure: Sensing Stress of Computer Users,
- [5] David Sun, Pablo Paredes, and John Canny, MouStress: Detecting Stress from Mouse Motion, EECS, University of California, Berkeley, 2014
- [6] Brindley, D., Rolland, Y., et al. Possible connections between stress, diabetes, obesity, hypertension and altered lipoprotein metabolism that may result in atherosclerosis. *Clinical science (London, England: 1979)* 77, 5 (1989), 453.
- [7] Hanson, P. *The Joy of Stress*. Pan Books, London, 1987
- [8] Roseman, R.H. & Chesney, M.A. (1985) Type A behaviour pattern: its relationship to coronary heart disease and its modification by behavioral and pharmacological approaches. In: *Stress in Health and Disease*, pp. 206-241. Ed. Zales, M.R. Bruner/Mazel Publishers, New York.
- [9] Sapolsky, R.M., Krey, L.C. & McEwen, B.S. (1986) The neuroendocrinology of stress and aging: the glucocorticoid cascade hypothesis. *Endocrine Review*, 7,284-301.
- [10] Python knjižnica »*pynput*«, <https://pypi.org/project/pynput/>, dostopno 25.5.2020
- [11] Allen B. Downey, *The Little Book of Semaphores*, 2009

- [12] Neema Moraveji, Charlton Soesanto, Towards Stress-less User Interfaces:10 Design Heuristics Based on the Psychophysiology of Stress, 2012
- [13] Ratna Sharma, Shveta Khera, Amit Mohan, Nidhi Gupta and Rooma Basu Ray, Assessment of Computer Game as a Psychological Stressor, New Delhi, 2006
- [14] S. Hey, A. Gharbi, B. v. Haaren, K. Walter, N. König and S. Löffler, "Continuous Noninvasive Pulse Transit Time Measurement for Psychophysiological Stress Monitoring," 2009 International Conference on eHealth, Telemedicine, and Social Medicine, Cancun, 2009, pp. 113-116, doi: 10.1109/eTELEMED.2009.35.