

I. FLUXBOARD – ADAPTIVNI INTERFEJS ZA UNOS TEKSTA

FLUXBOARD – ADAPTIVE TEXT INPUT INTERFACE

**ANDREJ TETKIĆ¹ IT41, ALEKSANDAR BALANESKO¹ IT41,
DIMITRIJE SAKAN¹ IT41**

Mentor **MILAN VUKOBRAT¹** dipl.ing.el. nastavnik stručnih predmeta

¹ Elektrotehnička škola “Mihajlo Pupin” Novi Sad

II. REZIME

Konvencionalne tastature potiču iz ere mehaničkih pisaćih mašina. Njihov fiksni QWERTY raspored predstavlja fundamentalno ograničenje koje zanemaruje anatomske razlike korisnika, primoravajući ljude da prilagođavaju prirodnu motoriku alatu. Usled dugotrajne upotrebe, to dovodi do neefikasnosti i ozbiljnih povreda poput sindroma karpalnog tunela. Kao odgovor, predlažemo inovativno rešenje: adaptivni kapacitivni senzorski interfejs za unos teksta. Naš sistem koristi veštačku inteligenciju za prevođenje slobodnih, višeprstnih gestova direktno u računarske komande. Uređaj kontinualno uči i prilagođava se specifičnoj motorici korisnika, ostvarujući superiornu, personalizovanu ergonomiju.

Arhitektura sistema koncipirana je kao jednosmeran protok podataka, počevši od akvizicije signala preko kapacitivne matrice i ESP32 mikrokontrolera. Izdvojeni pokreti prolaze kroz dvostruku VAE arhitekturu: konvolucionni Gesture VAE kompresuje prostorni raspored dodira, dok rekurentni Text VAE definiše kontinualni jezički prostor. Za sažimanje vremenske dimenzije dizajnirali smo sopstvenu temporalnu transformaciju linearne složenosti. Konačno prevođenje putanje u tekst oslanja se na ponderisano preslikavanje zasnovano na inverznoj udaljenosti (*Inverse Distance Weighting*) sa ugrađenim "magnetskim" svojstvima, što omogućava stabilnu predikciju i učenje bez zaboravljanja prethodnih obrazaca.

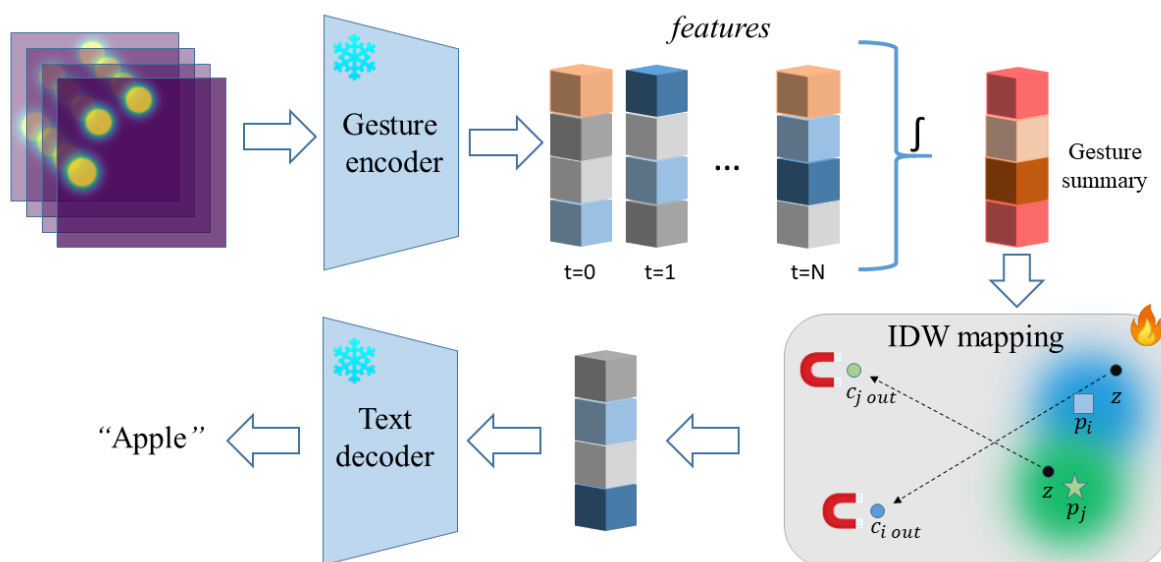
Trenutno, hardverski i softverski moduli nisu potpuno kompletirani. Izrada senzorskog prototipa nailazi na izazove usled strogih tolerancija pri laserskom sečenju, neophodnih za rad matrice. Ipak, preliminarna evaluacija uspešno validira koncept. Modeli postižu tačnost rekonstrukcije od 94,46% za Gesture VAE i preko 99% za n-grame do 10 karaktera kod Text VAE modela, pri čemu je u ovoj fazi prioritet dat brzini CPU inferencije umesto primeni kompleksnijih arhitektura.

III. PRINCIP RADA I PRAKTIČNA REALIZACIJA

A. Uvod i srodni radovi

Konvencionalne metode unosa teksta i dalje su fundamentalno ograničene nasleđem pisanih mašina, primoravajući korisnike na adaptaciju statičnim fizičkim ili virtuelnim QWERTY rasporedima. Iako su rešenja zasnovana na gestovima, poput SHARK-a (1) ili Swype-a, donela napredak, ona pate od kritičnih nedostataka: oslanjaju se na apsolutno pozicioniranje u odnosu na fiksnu vizuelnu pozadinu i inherentno su limitirana na putanju jednog prsta, čime se u potpunosti zanemaruje motorički potencijal cele šake. Ovakav pristup ne dozvoljava personalizaciju prema biometriji korisnika niti podržava intuitivne, simultane pokrete više prstiju. Kako bismo prevazišli ova ograničenja, predlažemo arhitekturu koja tretira unos kao problem prevođenja modaliteta, a ne pukog mapiranja koordinata. Inspirisan uspehom CLIP modela (2) u usklađivanju slike i teksta, naš sistem direktno projektuje višeprstne gestove u semantički prostor teksta, koristeći sopstvenu optimizovanu temporalnu transformaciju. Ovim se eliminiše potreba za predefinisanim tasterima i omogućava sistemu da kontinualno uči iz navika korisnika, organski spajajući česte n-grame radi veće fluidnosti i brzine.

Da bi se ostvarilo ovakvo dinamičko prevođenje, dizajnirali smo jednosmeran protok podataka (*pipeline*) koji sirove signale dodira transformiše u računarske komande, što je detaljno ilustrovano i objašnjeno na Slici 1.



sl. 1 - Arhitektura predloženog sistema za prevođenje gestova u tekst. Proces funkcioniše kao jednosmerni pipeline: (1) Kapacitivna površina registruje sekvencu sirovih signala (toplotne mape dodira). (2) Gesture VAE (konvolucionni enkoder) kompresuje svako prostorno stanje senzora u latentni vektor. (3) Niz ovih vektora ($t = 0$ do $t = N$) prolazi kroz optimizovanu temporalnu transformaciju ($\int$) koja sažima vremensku dimenziju pokreta u jedinstvenu tačku – Gesture summary (z). (4) Pomoću neparametarske interpolacije (IDW mapping), ova tačka se preslikava iz prostora gestova u tekstualni latentni prostor, oslanjajući se na poznate usidrene tačke (p_i, p_j). (5) Konačno, rekurentni dekodler (Text VAE) iz dobijene tačke rekonstruiše ciljani n-gram (npr. reč "Apple") i šalje ga operativnom sistemu.

B. Hardverska realizacija i pretprocesiranje signala

Primarni senzorski element konstruisan je u formi dvoosne matrice rezolucije 16×16 , koja se sastoji od 16 predajnih (Tx) i 16 prijemnih (Rx) linija. Elektrode su izrađene od aluminijumske trake i postavljene ortogonalno jedne u odnosu na druge u dva odvojena sloja.

Da bi se maksimizovao odnos signala i šuma (SNR) i povećala korisna površina za detekciju dodira, svaka linija je dizajnirana tako da se sastoji od niza od 16 proširenja u obliku dijamanta (romboida). Ovakva "dijamantska" topologija je standardna u dizajnu kapacitivnih ekrana jer obezbeđuje maksimalnu površinu međusobnog preklapanja polja na mestima ukrštanja Rx i Tx linija, dok se fizičko preklapanje samih provodnika na raskrsnicama minimizuje.

U cilju pružanja taktilne povratne informacije korisniku prilikom napuštanja aktivne zone senzora, obod uređaja je projektovan u formi graničnika visine 1 mm u odnosu na radnu površinu. Pored haptičke funkcije, ovaj element služi i kao mehaničko kućište koje omogućava sprovođenje i fiksaciju kontakata senzorske matrice sa upravljačkom elektronikom.

Aktuelna iteracija prototipa realizovana je na planarnoj podlozi. Ovakav pristup je usvojen radi optimizacije proizvodnog procesa, jednostavnije kalibracije senzorskog sistema i pouzdane validacije koncepta. Uprkos trenutnoj geometriji, predložena tehnologija poseduje visok potencijal za primenu na zakrivljenim površinama.

Arhitektura senzorskog uređaja zasnovana je na Multi-Touch Kit (3), koji detaljno definiše principe detekcije dodira. Umesto prvobitnog rešenja baziranog na Arduino platformi, korišćen je mikrokontroler ESP32. Ovakav izbor je uslovljen potrebom za višim radnim taktom procesora i većom rezolucijom analogno-digitalne konverzije (ADC) u odnosu na standardne Arduino ploče.

Povezivanje senzorske matrice ostvareno je na sledeći način:

Tx linije (Predaja): Provodnici sa 16 predajnih linija povezani su na 16-kanalni analogni multiplexer, koji se kontroliše putem digitalnih pinova ESP32 mikrokontrolera. Multiplexer omogućava sekvencijalno pobuđivanje svake Tx linije pojedinačno bez iscrpljivanja raspoloživih pinova na mikrokontroleru.

Rx linije (Prijem): Svaka od 16 prijemnih linija direktno se sprovodi do ADC pinova na ESP32. U delu graničnika, na svakoj Rx liniji nalazi se uzemljeni otpornik (*pull-down/load resistor*). Ovaj otpornik u kombinaciji sa kapacitivnošću senzora formira RC kolo. Dodir korisnika (prsta) menja lokalnu dielektričnu konstantu na preseku aktivne Tx i merene Rx linije, što rezultuje merljivom promenom nivoa napona prilikom očitavanja na ADC pinovima.

Princip rada firmvera zasniva se na tehnici vremenskog multipleksiranja (*time-multiplexed scanning*). Mikrokontroler šalje pobudni signal na prvu Tx liniju (kroz 16-kanalni

multiplekser), a zatim istovremeno (ili brzim sekvencijalnim očitavanjem) prikuplja analogne vrednosti sa svih 16 Rx linija putem ADC pinova. Ovaj proces se ponavlja za sve Tx linije, čime se generiše matrica kapacitivnosti od 256 preseka u jednom ciklusu osvežavanja (*frame*). Za potrebe ovog istraživanja, originalni softver je adaptiran specifikacijama ESP32 arhitekture. Prilagođavanja su obuhvatila promenu opsega i rezolucije analognog očitavanja – s obzirom na to da ovaj mikrokontroler nudi 12-bitni ADC, za razliku od 10-bitnog na klasičnim Arduino pločama – kao i optimizaciju tajminga očitavanja radi potpunog iskorišćenja njegove više radne frekvencije.

Prototip je projektovan da funkcioniše primarno kao interfejs za prikupljanje i prosleđivanje podataka, pri čemu se celokupna obrada i inferencija modela vrše eksterno na računaru. U tu svrhu, podaci se putem serijske komunikacije šalju u Processing okruženje, gde se vrši interpolacija i napredna obrada signala za mapiranje višestrukih dodira (*multi-touch tracking*). Istovremeno, važno je napomenuti da u tekućoj fazi razvoja senzorski uređaj nije u potpunosti kompletiran usled poteškoća u pronalaženju metode za precizno sečenje senzorskih jedinica, što predstavlja ključni preduslov za njegovu optimalnu izradu.

Sirovi podaci sa senzora dolaze u vidu dvodimenzionalne matrice kapacitivnih vrednosti R_t , analognih monohromatskoj slici pritiska, u trenutku t . S obzirom na to da je cilj sistema rad u realnom vremenu na hardveru ograničenih resursa, procesiranje se ne vrši kontinuirano nad svakim frejmom. Umesto toga, implementiran je mehanizam zasnovan na detekciji promene, tzv. *Flux*, koji aktivira dalju obradu samo kada promena signala pređe definisani prag. Flux Φ_t se definiše kao suma apsolutnih razlika između trenutnog i prethodnog stanja matrice:

$$\Phi_t = \sum_{i,j} |R_t[i,j] - R_{t-1}[i,j]| \quad (1)$$

Kada je uslov $\Phi_t > \theta_{flux}$ zadovoljen, sirovi podaci prolaze kroz fazu normalizacije. Prvi korak je izračunavanje centra mase (*Center of Gravity - CoG*) otiska, $(\bar{x}, \bar{y})$, koji služi kao referentna tačka za pozicioniranje. Kako bi se eliminisao šum senzora i mikro-pomeraji prsta koji nisu namerni, koordinate centra mase se propuštaju kroz *Low-Pass* filter pre dalje upotrebe.

Ključni korak u pretprocesiranju je transformacija sirove slike u reprezentaciju invarijantnu na poziciju. Koristeći izračunati CoG, generiše se polarna projekcija od 360 stepeni oko centra mase. Ovim postupkom se apsolutne koordinate pretvaraju u relativne, čime se postiže da sistem prepozna oblik šake bez obzira na to gde se ona nalazi na površini senzora. Ovako obrađen tenzor predstavlja ulaz u neuronske mreže.

C. Modeliranje latentnih prostora: Dvostruka VAE arhitektura

Konvencionalni pristupi prepoznavanju unosa često svode problem na diskretnu klasifikaciju

ili direktno mapiranje enkodiranih tačaka na stringove (npr. putem jednostavnog k-NN algoritma). Ovakve metode poseduju fundamentalni nedostatak: one tretiraju izlazni prostor kao skup nepovezanih simbola bez unutrašnje topologije, čime se gubi informacija o semantičkoj i morfološkoj bliskosti ciljanih reči. U takvom sistemu, greška modela je binarna, a direktno mapiranje ne dozvoljava interpolaciju između poznatih stanja, što čini sistem krutim i podložnim greškama usled varijacija u korisničkom unosu. Nasuprot tome, naša arhitektura formuliše problem kao online geometrijsku regresiju između dve visokodimenzionalne mnogostrukosti (*manifolds*). Koncept mapiranja jezika u kontinualni latentni prostor je prethodno istražen za generisanje teksta, ali primarno na semantičkom nivou celih reči i rečenica (4). Iako delimo ideju o jezičkom latentnom prostoru, naša arhitektura se fundamentalno razlikuje po tome što se fokusira na sekvence karaktera (n-grame) umesto na pojedinačne reči. Umesto klasifikacije, mi koristimo zamrznuti (*frozen*) Text VAE baziran na karakteristikama da definišemo kontinualni jezički prostor koji služi kao fiksna geometrijska meta. Ovakav pristup transformiše proces učenja iz memorisanja klasa u učenje glatkog preslikavanja između prostora gestova i prostora jezika. Ključna prednost korišćenja Text VAE modela, umesto direktnog string mapiranja, leži u geometrijskoj strukturi latentnog prostora: sekvence poput "mi" i "mo" se u ovom prostoru nalaze na merljivoj euklidskoj udaljenosti, što sistemu omogućava da kvantifikuje nesigurnost i vrši strukturiranu procenu pouzdanosti (*structured confidence estimation*). Ovo svojstvo omogućava "meko" (*smooth*) opadanje performansi pri perceptivnom driftu (promeni načina kucanja), umesto naglog otkaza karakterističnog za klasifikatore. Fundamentalno, ovakva arhitektura eliminiše problem katastrofalnog zaboravljanja (*catastrophic forgetting*) jer su težine oba VAE modela zamrznute, pa se kontinualna adaptacija korisniku ostvaruje isključivo kroz dinamičko dodavanje i ažuriranje mapiranja između dva latentna prostora, čime se novi gestovi usvajaju bez degradacije prethodno naučenog ponašanja. Time postizemo sistem koji je sposoban za kontinualnu adaptaciju bez degradacije osnovnih jezičkih sposobnosti, implementiran kroz dva lagana modela optimizovana za CPU inferenciju.

1. *Gesture VAE (Konvolucionni enkoder)*

Za enkodiranje prostornog rasporeda dodira koristimo arhitekturu Varijacionog Autoenkodera (VAE) (5) zasnovan na konvolucionim slojevima (CNN). Model je dizajniran da bude računarski efikasan (*lightweight*) kako bi se izvršavao na procesorima opšte namene (CPU) sa minimalnom latentnošću. Treniran je na sintetičkom skupu podataka od 100.000 generisanih slika koje simuliraju nasumične konfiguracije dodira (kako moguće, tako i nemoguće anatomske položaje). Ovakav *unsupervised* pristup primorava model da nauči robusne,

generalizovane karakteristike (*features*) prostora dodira.

Evaluacija na test setu pokazuje visoku preciznost rekonstrukcije, sa *Pixel Accuracy* od 94.46% (za piksele unutar ± 0.1 opsega) i *Mean Absolute Error* (MAE) od 0.0458. Ovi rezultati potvrđuju da 128-dimenzionalni latentni prostor uspešno kompresuje ključne informacije o obliku i orijentaciji dodira.

2. Text VAE (Rekurentni dekođer)

Ciljni geometrijski prostor definisan je pomoću rekurentnog VAE modela zasnovanog na GRU (*Gated Recurrent Unit*) ćelijama. Ovaj model uči da mapira sekvence karaktera (n-grama) u kontinuirani latentni prostor fiksne dimenzije (128D). Dataset za trening je dinamički generisan tokom učenja, sastojeći se od nasumičnih n-grama dužine 1 do 30 karaktera latiničnog alfabeta. S obzirom na ograničenja resursa u realnom vremenu, odabran je kompromis između veličine modela i tačnosti na dugim sekvencama. Rezultati (Tabela 1) pokazuju skoro savršenu rekonstrukciju (99.10%+ Accuracy) za n-grama dužine do 10 karaktera, što pokriva veliku većinu reči u prirodnom jeziku.

Tabela 1 – Rezultati validacije Text VAE na 10.000 nasumično generisanih n-grama

Length Range	Sample Size	Char Accuracy	Exact Match
1-5 chars	3043	99.99%	99.97%
6-10 chars	1775	99.75%	99.10%
11-15 chars	1703	96.26%	82.91%
16-20 chars	2484	76.11%	29.07%
21+ chars	995	43.19%	1.71%

Iako veći modeli koje smo testirali postižu bolju rekonstrukciju na dužim sekvencama, GRU arhitektura je zadržana zbog superiorne brzine inferencije, ključne za fluid osećaj kucanja. Ovakva postavka omogućava sistemu da "razume" tekst geometrijski – slične reči se nalaze blizu jedna drugoj u latentnom prostoru, što kasnije omogućava interpolaciju i ispravljanje grešaka.

D. Temporalna transformacija

Izlaz iz konvolucionog enkodera (Gesture VAE) predstavlja kontinualni tok latentnih reprezentacija trenutnih stanja senzora. S obzirom na to da se gestikulacija korisnika u realnom vremenu ne može unapred precizno segmentovati na zasebne akcije, sistem održava kružni bafer (*history buffer*) nedavnih stanja. Iz ove istorije, polazeći od trenutnog vremenskog preseka unazad, predlažemo dinamičko generisanje više preklapajućih vremenskih prozora (*time*

windows) različitog trajanja. Ovaj pristup omogućava sistemu da evaluira različite potencijalne dužine gesta i izdvoji željenu akciju iz neprekidnog signala.

Za svaki definisani vremenski prozor, sistem izdvaja sekvencu *embedding*-a dimenzionalnosti $(t \cdot D)$, gde je t broj vremenskih koraka, a $D = 128$ dimenzija latentnog prostora. Da bi se ova sekvenca mogla prevesti kroz Text VAE dekodera, neophodno je izvršiti sažimanje vremenske komponente u jedinstven vektor dimenzije (D) . Prilikom odabira metode dimenzionalne redukcije, konvencionalni duboki modeli (RNN, 1D CNN) odbačeni su zbog unošenja kašnjenja (*latency*) i narušavanja transparentnosti geometrijskog mapiranja. Sa druge strane, matematičke metode poput signaturne transformacije (*Signature Transform*) (6, 7) pokazale su se kao previše računarski zahtevne. Dok trankirane signaturne transformacije skaliraju geometrijski sa dubinom, tipično zahtevajući vreme izvršavanja $O(N \cdot D^k)$ za signaturu dubine k , naš sistem izračunava ovu transformaciju višestruko u svakom vremenskom koraku (za svaki generisani preklapajući prozor), zahtevajući drastično manju asimptotsku složenost. Zbog toga smo dizajnirali sopstvenu laganu temporalnu transformaciju koji zadovoljava rigorozne kriterijume: kontinuitet, zavisnost od smeru, visoku računarsku efikasnost i invarijantnost na gustinu uzorkovanja tačaka.

1. Matematička formulacija modela

Kako bismo mapirali putanje varijabilne dužine i nepravilnog uzorkovanja u reprezentacije fiksne dimenzije, sekvencu formulišemo kao deo-po-deo linearnu krivu (*piecewise linear curve*) i vršimo evaluaciju kontinualnog linijskog integrala nad njenom normalizovanom dužinom luka.

Neka je ulazna sekvenca od N *embedding*-a predstavljena kao uređeni skup tačaka u D -dimenzionalnom prostoru, $P = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_N\}$, gde je $\mathbf{p}_i \in \mathbb{R}^D$. Definišemo kontinualnu putanju $\mathbf{p}(s)$ formiranu linearnom interpolacijom skupa P , parametrizovanu njenom apsolutnom dužinom luka $s \in [0, L]$. Ukupna dužina putanje L data je sumom Euklidskih rastojanja između uzastopnih tačaka:

$$L = \sum_{i=1}^{N-1} \|\mathbf{p}_{i+1} - \mathbf{p}_i\|_2 \quad (2)$$

Da bi se osigurala invarijantnost na apsolutnu skalu sekvence (čime se anulira uticaj brzine kojom korisnik izvodi gest), vršimo reparametrizaciju putanje koristeći normalizovanu promenljivu dužine luka $t = s/L$, skalirajući domen na $t \in [0, 1]$. Korak enkodiranja projektuje ovu putanju u fiksnu reprezentaciju $\mathbf{z} \in \mathbb{R}^D$ rešavanjem integrala putanje po elementima (*element-wise*), modulisanog skupom statičkih harmonijskih baznih funkcija:

$$\mathbf{z} = \int_0^1 \mathbf{p}(t) \odot \cos(\omega t + \phi) dt \quad (3)$$

gde $\odot$ označava Adamarov (*Hadamard*) proizvod. Bazne funkcije deluju kao fiksni, višeskalarni pozicioni lenjir (*multi-scale positional ruler*). Kako bi se obuhvatila i široka geometrijska struktura gesta i mikro-varijacije, frekvencije $\omega \in \mathbb{R}^D$ se uzorkuju iz log-uniformne distribucije $\omega_d \sim \exp(\mathcal{U}(\ln(\omega_{min}), \ln(\alpha \cdot \omega_{max})))$, gde α deluje kao hiperparametar koji kontroliše osetljivost na visoke frekvencije. Fazni pomaci $\phi \in [0, 2\pi]^D$ se uzorkuju uniformno kako bi se održala ortogonalnost kroz dimenzije.

U praksi, ova kontinualna formulacija se evaluira direktno na diskretnoj ulaznoj sekvenci korišćenjem numeričke integracije putem pravila srednje tačke (*midpoint rule*). Za svaki segment $i \in \{1, \dots, N-1\}$ oivičen tačkama $(\mathbf{p}_i, \mathbf{p}_{i+1})$, definišemo njegovu Euklidsku dužinu $l_i = \|\mathbf{p}_{i+1} - \mathbf{p}_i\|_2$, što daje frakciono vremensko trajanje $\Delta t_i = l_i/L$. Prostorna srednja tačka segmenta je $\bar{\mathbf{p}}_i = (\mathbf{p}_i + \mathbf{p}_{i+1})/2$, koja se javlja u normalizovanoj vremenskoj srednjoj tački $\bar{t}_i = \frac{1}{L}(\sum_{j=1}^{i-1} l_j + \frac{l_i}{2})$. Konačna diskretna aproksimacija integrala je:

$$\mathbf{z} \approx \sum_{i=1}^{N-1} \bar{\mathbf{p}}_i \odot \cos(\omega \bar{t}_i + \phi) \Delta t_i \quad (4)$$

Ključna prednost ove formulacije leži u njenoj računarskoj efikasnosti. Za razliku od trankiranih signaturnih transformacija, ovaj pristup integrala putanje zadržava striktnu linearnu složenost $O(N \cdot D)$. Ovo omogućava izuzetno efikasno enkodiranje visokodimenzionalnih putanja bez eksponencijalne eksplozije karakteristika (*feature explosion*) koja prati tenzorske metode višeg reda, pritom garantujući potpunu invarijantnost na gustinu uzorkovanja sirovog signala.

E. Dinamično mapiranje i predikcija

Proizvod temporalne transformacije, vektor $\mathbf{z}$, predstavlja sažetu reprezentaciju putanje. Neophodno je ovaj vektor preslikati iz njegovog prostora u ciljni jezički prostor. Za ovu svrhu smo nezavisno izveli specifičnu formulaciju neparametarske interpolacije, projektovanu da osigura stabilnost predikcije u uslovima varijabilnosti ulaza. Iako matematička struktura težinskih koeficijenata odgovara Shepardovoj metodi (*Inverse Distance Weighting*) [8], naš pristup se razlikuje u primeni lokalnih ofseta kako bi se postigao efekat „magnetskog” stabilizovanog preslikavanja.

Sistem se oslanja na bazu podataka koja sadrži usidrene tačke (*anchor points*), gde je svaki unos par $(\mathbf{p}_i, \mathbf{c}_i)$ koji definiše poznato preslikavanje između ulazne putanje $\mathbf{p}_i$ i njenog ciljnog tekstualnog *embedding*-a $\mathbf{c}_i$. Za bilo koji novi ulaz $\mathbf{z}$, njegova korespondentna izlazna

tačka $\mathbf{c}_{out}$ se izračunava putem ponderisane interpolacije na osnovu k najbližih suseda. Težinski koeficijenti w_j , koji određuju uticaj svakog suseda, računaju se korišćenjem inverzne euklidske udaljenosti podignute na stepen s :

$$w_j = \frac{\frac{1}{\|\mathbf{z} - \mathbf{p}_j\|^s}}{\sum_{i=1}^k \frac{1}{\|\mathbf{z} - \mathbf{p}_i\|^s}} \quad (5)$$

Hiperparametar s , nazvan parametar snage (*power parameter*), kontroliše lokalnost uticaja. Korišćenjem visokih vrednosti (npr. $s \geq 5$), uticaj udaljenijih suseda postaje zanemarljiv, čime se interpolacija efektivno svodi na lokalnu deformaciju prostora pod dominantnim uticajem najbližih tačaka. Konačni izlazni vektor $\mathbf{c}_{out}$ dobija se primenom ponderisane sume translacionih vektora $(\mathbf{c}_i - \mathbf{p}_i)$ na originalni ulaz $\mathbf{z}$:

$$\mathbf{c}_{out} = \mathbf{z} + \sum_{i=1}^k w_i (\mathbf{c}_i - \mathbf{p}_i) \quad (6)$$

Ovakva formulacija je odabrana kako bi se postigla visoka robusnost sistema. Čak i kada ulazni vektor $\mathbf{z}$ nije identičan usidrenoj tački $\mathbf{p}_i$, dodavanje interpoliranog ofseta omogućava da izlaz „teži” ka ispravnom *embedding*-u $\mathbf{c}_i$. Time se postiže efekat stabilizacije (*magnetic mapping*), gde sistem zadržava preciznost mapiranja uprkos nesavršenostima u reprezentaciji putanje, dok istovremeno zadržava prednosti potpune interpolacije u prelaznim stanjima.

U praksi, baza podataka sa parovima $(\mathbf{p}_i, \mathbf{c}_i)$ je indeksirana prostornom strukturom podataka Ball Tree (9), koja omogućava efikasno pronalaženje k najbližih suseda.

Moć ovakvog pristupa postaje očigledna u kontekstu kontinualne obrade signala. Za svaki od paralelnih vremenskih prozora, sistem izvodi ovu interpolaciju i računa nivo pouzdanosti (*confidence*) predikcije, koji je proporcionalan težini najbližeg suseda. U svakom trenutku, sistem prikazuje izlaz prozora sa trenutno najvišim nivoom pouzdanosti. Ključna inovacija je u tome što sistem ne donosi konačnu odluku odmah. Ako se u budućem vremenskom koraku pojavi duži vremenski prozor, koji obuhvata prethodni ali daje predikciju sa većim ukupnim *confidence*-om (logaritamski skaliranim sa dužinom prozora), on će retroaktivno zameniti prethodnu, kraću predikciju. Ovaj mehanizam omogućava sistemu da efektivno vrši "simuliranu segmentaciju" (*simulated segmentation*) u realnom vremenu, dinamički odlučujući o granicama gesta na osnovu najverovatnijeg tumačenja u širem kontekstu.

F. Računarski interfejs i adaptivno učenje

Konačni izlaz sistema se realizuje kroz softverski interfejs koji emulira rad standardne tastature, osiguravajući kompatibilnost sa bilo kojim postojećim softverom. U svakom trenutku kada se

promeni najverovatnija predikcija iz skupa aktivnih vremenskih prozora, interfejs tu promenu dinamički prenosi operativnom sistemu. Ukoliko se radi o retroaktivnoj ispravci, gde duži gest redefiniše značenje prethodnog kraćeg gesta, interfejs prvo šalje komande za brisanje (*backspace*) pogrešno interpretiranih karaktera, a zatim ispisuje novu, tačniju sekvencu. Ovaj mehanizam stvara fluidno iskustvo kucanja koje dozvoljava "ispravke u letu".

Kako bi se ostvarila personalizacija i povećala preciznost tokom vremena, predlažemo implementaciju zatvorene petlje adaptivnog učenja (*closed-loop adaptive learning*). Ovaj mehanizam bi omogućio sistemu da u realnom vremenu uči iz eksplicitnih i implicitnih povratnih informacija korisnika. Kada korisnik manuelno ispravi pogrešno protumačeni gest, sistem bi tu ispravku tretirao kao supervizijski signal. Željeni tekstualni izlaz bi se enkodirao pomoću zamrznutog Text VAE enkodera kako bi se dobio ciljni *embedding* $\mathbf{c}_{target}$. Originalna sekvencu *embedding*-a koja je dovela do greške, nakon prolaska kroz temporalnu transformaciju, dala bi ulazni vektor $\mathbf{p}_{source}$. Sistem bi zatim kreirao ili ažurirao usidrenu tačku $(\mathbf{p}_{source}, \mathbf{c}_{target})$ u Shepardovoj interpolacionoj bazi, čime bi se prostor lokalno "ispravio" za buduće slične gestove.

Pored eksplicitnih ispravki, predlažemo i mehanizam pasivnog učenja. Ukoliko sistem generiše predikciju sa niskom pouzdanošću, ali korisnik ne izvrši ispravku, to se tumači kao implicitna potvrda. U tom slučaju, postojeća relacija mapiranja u interpolacionoj bazi bi se ojačala, povećavajući pouzdanost za buduće slične unose. Vremenom, ovaj proces kontinualne modifikacije prostora preslikavanja omogućava sistemu da se prilagodi jedinstvenoj motorici i navikama svakog korisnika.

Dodatno, predlažemo mehanizam za spajanje gestova (*gesture merging*). Analizom frekvencije uzastopnih n-grama, sistem bi mogao da identifikuje sekvence koje se često pojavljuju zajedno. Korisniku bi se implicitno omogućilo da te sekvence spoji u jedan, fluidniji pokret. Sistem bi naučio da prepozna ovu novu, dužu putanju i direktno je mapira na spojeni n-gram, omogućavajući korisniku da nesvesno razvija sopstvene skraćenice i time dramatično poveća brzinu unosa.

Kako bi se korisniku olakšao početak korišćenja, predlažemo razvoj prateće aplikacije za inicijalizaciju (*onboarding*). Ova aplikacija bi vodila korisnika kroz proces unosa predefinisanih reči i fraza. Svaki uspešan unos bi se koristio za kreiranje početnog skupa usidrenih tačaka u interpolacionoj bazi, čime bi se sistem inicijalno kalibrisao na osnovnu motoriku korisnika i postao upotrebljiv znatno brže nego sa potpuno praznim početnim stanjem.

IV. REZULTAT

Evaluacija predložene arhitekture sprovedena je na dva nivoa: kroz kvantitativnu analizu

performansi pojedinačnih softverskih modula i kroz kvalitativnu procenu trenutnog stanja integrisanog sistema.

1. Performanse Modela i Mehanizma Mapiranja

Osnovni cilj implementacije VAE modela bio je validacija arhitekturnog koncepta, a ne postizanje vrhunske tačnosti rekonstrukcije. S tim u vidu, trenutni modeli – konvolucionni Gesture VAE sa Pixel Accuracy od 94.46% i rekurentni Text VAE sa preko 99% tačnom rekonstrukcijom za n-grame do 10 karaktera – ispunili su svoju ulogu kao funkcionalne komponente sistema. Arhitekture koje omogućavaju superiorne performanse su poznate i biće implementirane u narednim iteracijama, ali za ovu fazu demonstracije, prioritet je dat brzini inferencije na CPU-u.

Validacija temporalne transformacije i Shepardove interpolacije izvršena je kroz eksperiment učenja sa jednim primerom (*one-shot learning*). U ovom eksperimentu, 51 jedinstveni simbol je unet pojedinačno, čime je kreirana inicijalna baza usidrenih tačaka. Naknadna evaluacija je pokazala visoku uspešnost, sa tačnošću klasifikacije od 98% (50 od 51 simbola je tačno prepoznato). Ključno zapažanje je da sistem nije pokazao degradaciju performansi niti interferenciju sa prethodno unetim simbolima sa povećanjem baze, što potvrđuje skalabilnost neparametarskog pristupa. Sistematska greška je detektovana isključivo pri razlikovanju geometrijski sličnih simbola "7" i ">", što ukazuje da je za fine distinkcije potrebno dalje usavršavanje faze preprocesiranja. Važno je naglasiti da je ova evaluacija koristila simplifikovani, jednoprotlačni unos (simulacija crtanja jednim prstom) radi validacije rada osnovanog matematičkog modela, dok je arhitektura dizajnirana za znatno kompleksniji, višeprstni unos koji uključuje celu šaku.

2. Kvalitativna Evaluacija i Ograničenja Prototipa

Trenutni sistem funkcioniše kao softverski prototip koji potvrđuje ključne koncepte, ali sa jasno definisanim hardverskim i arhitekturnim ograničenjima.

Hardverska ograničenja: Razvoj ciljanog hardvera – kapacitivne matrice visoke rezolucije – predstavlja značajan proizvodni izazov. Metode izrade se aktivno istražuju, ali postizanje izuzetno malih tolerancija neophodnih za pouzdan rad još uvek nije dostignuto. Usled toga, za prikupljanje podataka je korišćen touchpad uređaja MacBook Pro kao zamenski senzor, čiji su izlazni podaci o poziciji i pritisku transformisani da simuliraju sirov signal analogan preciznošću naše moguće implementacije senzora. Celokupna obrada podataka se trenutno izvršava na *host* računaru, dok ESP32 mikrokontroler služi isključivo kao relej.

Arhitekturna ograničenja: Predloženi mehanizam retroaktivne promene predikcije je primenjiv isključivo na unos teksta i ne podržava nepovratne akcije (npr. sistemske komande),

za koje bi bio potreban poseban mehanizam potvrde. Pored toga, ključne funkcionalnosti koje sistem čine istinski adaptivnim – mehanizam za aktivno menjanje usidrenih tačaka i dinamička segmentacija putem paralelnih vremenskih prozora – još uvek su u fazi teorijskog modela i nisu implementirane. Kompletiranje ovih modula predstavlja osnovu za budući rad.

V. ZAKLJUČAK

Ovaj rad je adresirao fundamentalna ograničenja tradicionalnih ulaznih uređaja i HCI (*Human-Computer Interaction*) paradigmi, pre svega njihovu rigidnost i nametanje "one-size-fits-all" prostornih rasporeda. Predstavili smo i evaluirali konceptualnu hardversko-softversku arhitekturu koja u potpunosti eliminiše potrebu za fiksnim layout-om. Definisanjem problema unosa kao online geometrijske regresije između prostora gestova i prostora jezika, razvili smo sistem koji se oblikuje prema korisniku, a ne obrnuto. Kroz inovativnu primenu dvostrukog VAE sistema, lagane temporalne transformacije i Shepardove interpolacije, demonstrirana je mogućnost kreiranja interfejsa u kojem su sve komande i tekstualni unosi isključivo definisani od strane korisnika. Ovakav pristup omogućava organski rast fluidnosti i brzine interakcije sa računarom, srazmerno vremenu koje korisnik uloži u sistem.

Pored očiglednih arhitekturnih inovacija, šire implikacije ovog rešenja zadiru duboko u domen pristupačnosti (*accessibility*). S obzirom na to da sistem ne zahteva precizno gađanje predefinisanih tastera, već prepoznaje relativnu geometriju slobodnih pokreta šake, on predstavlja potencijalno revolucionarno rešenje za osobe sa specifičnim motoričkim ograničenjima, hroničnim bolovima u zglobovima ili drugim fizičkim stanjima koja onemogućavaju korišćenje standardne tastature, a koje i dalje poseduju mobilnost ruku. Fleksibilnost ovakvog interfejsa vraća kontrolu korisniku, omogućavajući mu da kreira sopstveni biomehanički rečnik pokreta koji mu najviše odgovara.

Kako bi se ovaj teorijski i prototipski model preveo u potpuno zreo proizvod, budući rad će prvenstveno biti fokusiran na rešavanje trenutnih ograničenja. To podrazumeva izradu pouzdane kapacitivne matrice visoke rezolucije, implementaciju arhitekture paralelnih vremenskih prozora za dinamičku segmentaciju u realnom vremenu, kao i aktivaciju predložene petlje za adaptivno učenje (*closed-loop learning*). Pored toga, dalja unapređenja sistema obuhvatiće integraciju naprednih NLP modela za autocorrect funkcionalnost, koja bi korigovala semantičke greške na najvišem nivou. Iako predložena arhitektura po svojoj prirodi podržava simultani unos (*chorded input*), planira se njeno strukturno rafinisanje kako bi se prirodno i eksplicitno podržalo spajanje kompleksnih gestova više prstiju odjednom. Takođe, razvojni plan obuhvata proširenje sistema na dvoručni rad (*bimanual input*) i dizajniranje senzorskih površina različitih, nekonvencionalnih geometrijskih oblika, čime bi se maksimalno iskoristio

ergonomski potencijal tehnologije.

Konačni cilj budućih istraživanja je izrada potpuno integrisanog, nezavisnog uređaja (*on-device processing*) na kojem će se sprovesti opsežne korisničke studije (*user studies*). Prikupljanje empirijskih podataka sa većom grupom ispitanika biće neophodno kako bi se precizno izmerila kriva učenja (*learning curve*), maksimalna brzina unosa i zamor korisnika tokom produžene upotrebe. Tek kroz ovakvu, rigoroznu i detaljnu evaluaciju svih pojedinačnih podsistema i ukupnog korisničkog iskustva, biće moguće kvantifikovati pravu vrednost ovog interfejsa u poređenju sa viševjekovnim nasleđem tradicionalnih tastatura.

VI. LITERATURA I REFERENCE

- [1.] S. Zhai and P.-O. Kristensson, "Shorthand writing on stylus keyboard," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, in CHI '03*. New York, NY, USA: Association for Computing Machinery, Apr. 2003, pp. 97–104.
- [2.] A. Radford et al., "Learning Transferable Visual Models From Natural Language Supervision," in *Proceedings of the 38th International Conference on Machine Learning, PMLR*, Jul. 2021, pp. 8748–8763.
- [3.] N. Pourjafarian, A. Withana, J. A. Paradiso, and J. Steimle, "Multi-Touch Kit: A Do-It-Yourself Technique for Capacitive Multi-Touch Sensing Using a Commodity Microcontroller," in *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology, in UIST '19*. New York, NY, USA: Association for Computing Machinery, Oct. 2019, pp. 1071–1083.
- [4.] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Jozefowicz, and S. Bengio, "Generating Sentences from a Continuous Space," May 12, 2016.
- [5.] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," Dec. 20, 2013.
- [6.] I. Chevyrev and A. Kormilitzin, "A Primer on the Signature Method in Machine Learning," 2016, pp. 3–64.
- [7.] P. Bonnier, P. Kidger, I. P. Arribas, C. Salvi, and T. Lyons, "Deep Signature Transforms," in *Advances in Neural Information Processing Systems*, May. 21, 2019.
- [8.] D. Shepard, "A two-dimensional interpolation function for irregularly-spaced data," in *Proceedings of the 1968 23rd ACM national conference, in ACM '68*. New York, NY, USA: Association for Computing Machinery, Jan. 1968, pp. 517–524.
- [9.] S. Omohundro, "Five Balltree Construction Algorithms," 1989.