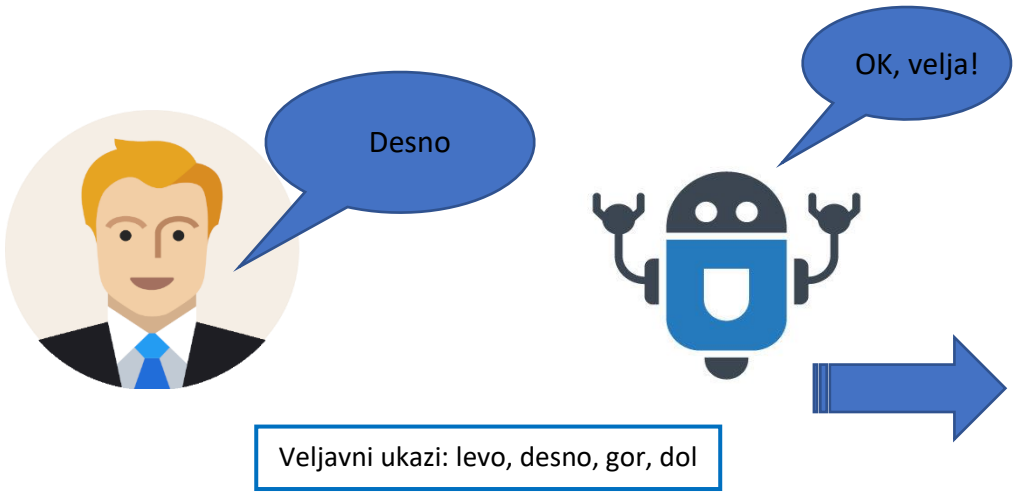
LABORATORIJSKA VAJA 2

IDEJA VAJE

Namen vaje je seznaniti študente s postopki razpoznavanja govora preko implementacije sistema za razpoznavanje kratkih govornih ukazov, ki temelji na predstavitvi govora s kepstralnimi značilkami, primerjavo vzorcev preko ukrivljanja časovne osi, ter razvrščanjem vzorcev po pravilu najbližjega soseda.

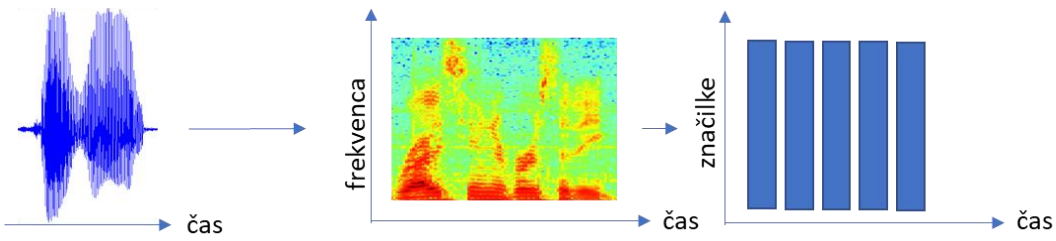
V okviru vaje bomo poskušali udejanjiti preprosto programsko rešitev, s katero je tudi v realnosti mogoče govorno upravljati z napravami. Problem bomo poenstavili na proprosto nalogo razpoznavanja štirih ločeno izgovorjenih ukazov. Primer aplikacije, kjer bi takšno razpoznavanje lahko bilo uporabno je govorno vodenje robotov, kot je prikazano na spodnji sliki.



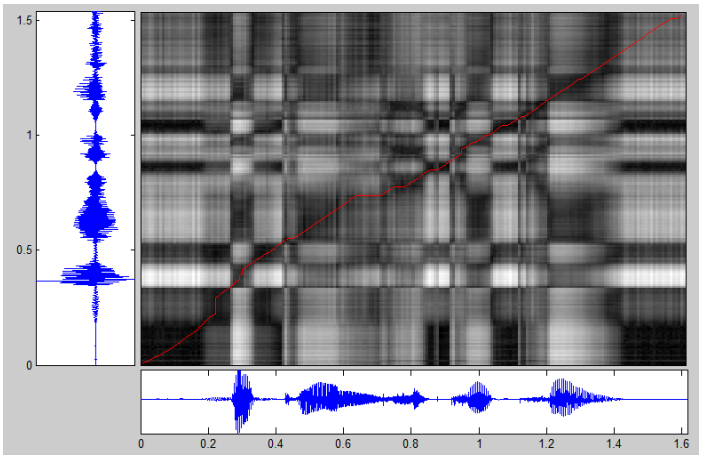
TEORETIČNO OZADJE

Sistem za razpoznavanje kratkih govornih ukazov, ki ga bomo udejanjili v okviru te vaje, je sestavljen iz dveh kritičnih komponent, in sicer iz parametrizacije odsekov govora, der določitve poravnave in razdalje med parametriranimi odseki, kot je prikazano na naslednjih slikah.

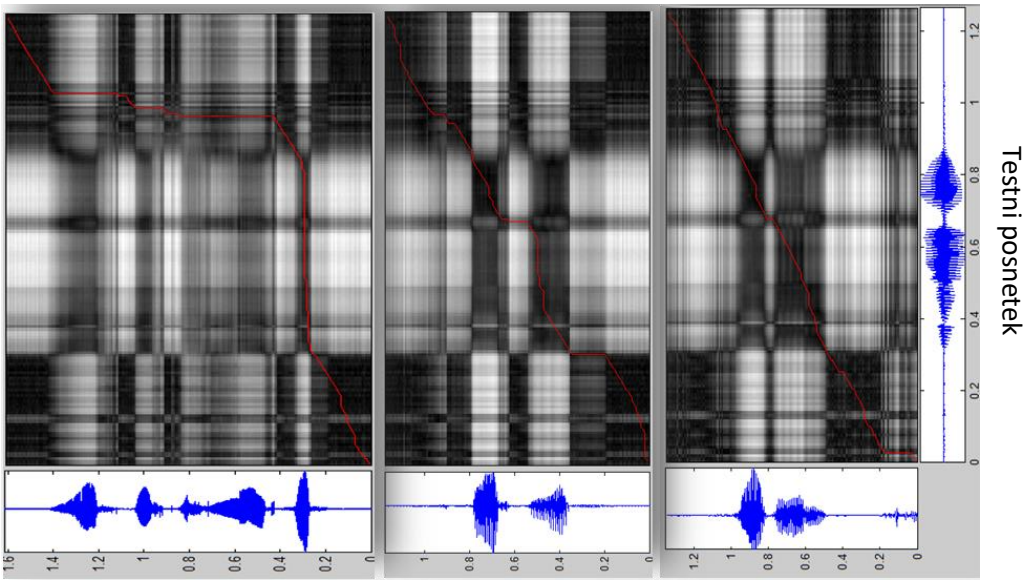
Korak 1: Izpeljava značilk (koeficientov MFCC iz govornih signalov) – parametrizacija signalov (Naloga 1)



Korak 2: Primerjava signalov z DTW - določitev podobnosti/razdalje med signaloma (Naloga 2)



Korak 3: Določitev izgovorjenega ukaza na podlagi največje podobnosti - razpoznavanje (Naloga 3 in 4)
OPOZORILO: Spodnja slika je rotirana zaradi velikosti (90° v nasprotni smeri ure). Optimalno poravnava še zmeraj poteka od levega spodnjega do desenga zgornjega kota.



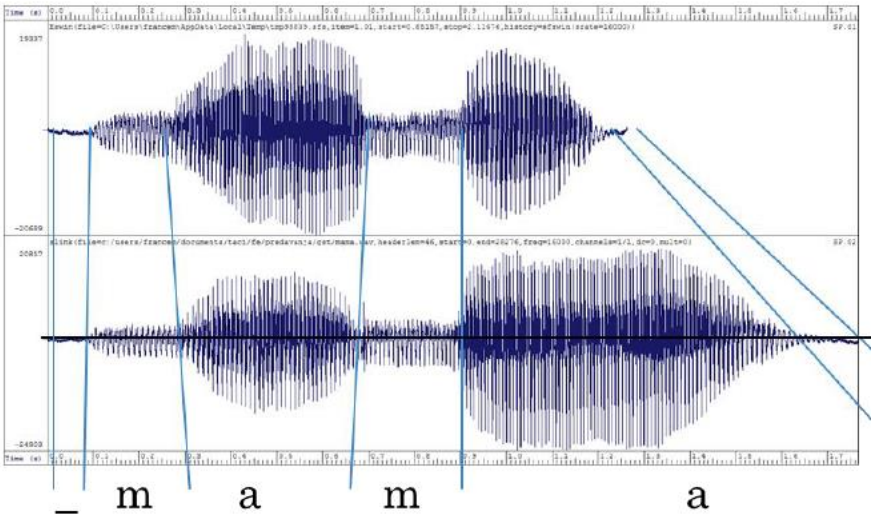
Prototip 1 (npr. levo) Prototip 2 (npr. gor) Prototip 3 (npr. dol)

RAZLAGA IN POVEZAVE NA TEORETIČNE OSNOVE

Korak 1: Parametrizacija odsekov govora je metoda za izpeljavo numeričnih značilk, ki nam omogočajo določitev izgovorjenega glasu na posameznem odseku govora. V ta namen bomo tu uporabili značilke kepstralnih koeficientov v melodični lestvici (angl. Mel-Frequency Cepstrum Coefficients, MFCC), ki nam podajo kratkočasovno moč govornega signala v frekvenčnih območjih, najpomembnejših za razpoznavanje govora. Natančnejše informacije o izpeljavi MFCC

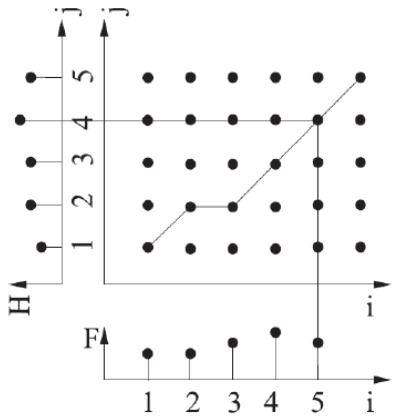
značilk ste si ogledali na predavanjih v okviru predavanja 4. Opis postopka najdete na drsnicah, objavljenih v e-učilnici.

Korak 2: Poravnava govornih odsekov je potrebna za natančno razpoznavanje, kadar pride do razlike v hitrosti izgovorjave posameznih besed med govorci. Algoritem dinamičnega ukrivljanja časovne osi (angl. Dynamic Time Warping, DTW) nam (ob danih omejitvah algoritma) določi optialno poravnavo med parom signalov kot je prikazano na spodnjem primeru. Primer prikazuje dva govorna signala, ki oba vsebujeta besedo „mama“. Zaradi različnih hitrosti izgovorjave in drugih dejavnikov variabilnosti je potrebno za določitev podobnosti signala med sabo poravnati. To dosežemo s postopkom dinamičnega ukrivljanja časovne osi, ki nam poleg poravnave vrne še ceno poravnave (oz. mero podobnosti), ki jo lahko uporabimo za razpoznavanje.



Predpostavimo, da na sliki zgornji signal predstavlja prototip (v naprej posneti signal z znano gvorno vsebino) in spodnji signal predstavlja testni govorni ukaz, ki ga želimo razpoznati. Z dinamičnim ukrivljanjem časovne osi določimo, kateri odsek testnega posnetka sovpada s katerim odsekom prototipa. Pri poravnavi sta tako prototip kot testni signal predstavljena v obliki zaporedja vektorjev MFCC značilk, ki določajo posamezne odseke teh posnetkov. Poravnava in določitev podobnosti se torej izvaja na nivoju zaporedja značilk in ne na časovnih signalih. Pred uporabo DTW je zato potreben Korak 1, ki nam vse posnetke pretvori v zaporedje MFFCC značilk.

Naj bo predstavitev prototipnega posnetka zaporedje H in predstavitev testnega posnetka zaporedje F kot je prikazano na spodnji sliki. Poravnavo časovne osi F glede na H lahko prikažemo s sledečim diagramom:



Na horizontalni osi, indeksirani z i, so odseki signala F (oz. njihove predstavitve z MFCC značilkami), na vertikalni osi, indeksirani z j pa odseki signala H. Cilj algoritma DTW je najti najcenejšo pot od točke $(i,j) = (1,1)$ (tj., prvega odseka obeh signalov) do točke $(i,j) = (|F|,|H|)$, kjer se oba signala končata. Cena vsake od točk (i,j) v grafu pa je določena z razdaljo med vektorjema značilk F_i in H_j v posamezni točki. Ker je po začetni oceni cen vsake izmed točk (i,j) potrebno piskati še najcenejšo pot skozi graf, kar je lahko računsko zahtevno, v praksi omejimo dovoljene prehode skozi graf. Možne omejitve ste predsatvili na predavanjih. Vsota cen po najdeni najcenejši poti nam nato predstavlja razdaljo D_{DTW} med signaloma.

Teoretično podlago za ta del vaje smo spoznali na predavanjih 5. Drsnice predavanj so objavljene v e-učilnici.

Korak 3: Razdaljo med signaloma lahko uporabimo za razpoznavanje govora preko uporabe prototipnih ukazov, kjer so prototipni ukazi v naprej posneti govorni signali z znano vsebino (npr. posnetki ukazov: gor, dol, levo, desno). Posnamemo več primerov vsakega izmed ukazov, ki ga želimo razpoznati, ter izračunamo parametrizacijo vsakega posnetka z značilkami MFCC. Ko želimo razpoznati nov (neznani, testni) signal, ga najprej pretvorimo v zaporedje vektorjev značilk (korak 1, parametrizacija), ter nato določimo razdaljo D_{DTW} do vsakega izmed v naprej posnetih prototipnih signalov. Po pravilu razvrščanja najbližjih sosedov nov signal razpoznamo kot tisti ukaz, katerega prototip ima najmanjšo razdaljo do novega posnetka. Ker je vsebina prototipnih ukazov znana z določitvijo najbolj podobnega prototipa tstnemu posnetku določimo tudi vsebino testnega posnetka.

Teoretično podlago za ta del vaje smo spoznali na predavanjih 5. Drsnice predavanj so objavljene v e-učilnici.

IZVEDBA

Za izvedbo vaje imajo študentje poleg že znanih programskih knjižnic numpy, scipy in matplotlib.pyplot na voljo še knjižnjico python_speech_features, ki je priložena dodatnemu gradivu vaje. Relevantne funkcije se iz knjižnjice uvozijo z ukazom

```
from python_speech_features import mfcc, delta
```

Namen vaje je z obstoječo knjižnjico za izpeljavo govornih značilk iz akustičnih signalov ter z mero razdalje med dvema zaporedjima značilk, ki jo definira postopek ukrivljanja časovne osi udejanjiti preprost razpoznavalnik kratkih (enobesednih) govornih ukazov.

Naloga 1 (1 točka)

Preučite dokumentacijo knjižnjice python_speech_features. Pri tem se osredotočite na funkciji python_speech_features.mfcc ter python_speech_features.delta. Opišite pomen vseh vhodnih argumentov funkcij. Spišite funkcijo mfcc_znacilke(signal, fs), ki naj izračuna MFCC značilke podanega signala s podano frekvenco vzorčenja. Pri tem računajte 12 koeficientov MEL kepstra ter logaritem kratkočasovne glasnosti (energije) signala. Tem 13 značilkam dodajte tudi njihove pripadajoče dinamične značilke prvega in drugega reda, tako, da bo vsak odsek signala predstavljen z vektorjem značilk z 39 elementi. Za izračun predstavitve z MFCC značilkami uporabite odseke signala dolžine 25 ms, s prekrivanjem 10 ms.

Funkcijo lahko preizkusite s signalom „gor1.wav“, ki se nahaja v mapi „posnetki“. Rezultat bi moral biti blizu zaporedja vektorjev značilk, shranjenega v datoteki mfcc_test.npy. To lahko preverite s sledečo skripto:

```
import numpy as np
from python_speech_features import mfcc, delta
from scipy.io.wavfile import read
import sys, os

def mfcc_feats(sig, fs): # implementiraj
    raise NotImplementedError

if __name__ == "__main__":
    fs, signal = read("posnetki/gor1.wav")
    dejanske_znacilke = mfcc_feats(signal, fs=16000)
    pravilne_znacilke = np.load("mfcc_test.npy")

    # primerjava oblike zaporedij vektorjev značilk - to bi se moralo ujemati
    print("Pravilna oblika:", pravilne_znacilke.shape)
    print("Dejanska oblika:", dejanske_znacilke.shape)

    # največje odstopanje med vašim izračunom
    # in dejanskim rezultatom -
    # to bi moralo biti blizu oz. enako 0

    print("max. odstopanje:",
          np.abs(pravilne_znacilke - dejanske_znacilke).max())
```

Testna skripta je na voljo tudi v dodatnem gradivu, gl. datoteko mfcc_test.py .

O teoretično ozadju tega dela vaje si lahko prebereti v drsnicah Predavanj 4.

Naloga 2 (2 točki)

Spišite funkcijo dtw_dist, ki naj izračuna razliko med dvema signaloma na podlagi ukrivljanja časovne osi njunih predstavitev z MFCC značilkami. Naj bosta to signal F, predstavljen z zaporedjem vektorjev značilk $(f_1, f_2, \dots, f_P)$, in signal H, predstavljen z zaporedjem vektorjev značilk $(h_1, h_2, \dots, h_R)$. Njuno razdaljo preko ukrivljanja časovnih osi, $D_{dtw}(f, g)$ lahko izračunamo po algoritmu:

$C \leftarrow matrika\ velikih\ števil\ velikosti\ (P+1) \times (R+1)$

$C_{1,1} \leftarrow 0$

$za\ i := 1 \dots P$:

$za\ j := 1 \dots R$:

$C_{min} := \min\{C_{i,j}, C_{i+1,j}, C_{i,j+1}\}$

$C_{i+1,j+1} := C_{min} + \|f_i - h_j\|_2$

$konec\ za - stavka$

$konec\ za - stavka$

$D_{dtw} = C_{P,R}$

Kjer izraz $\|f_i - h_j\|_2$ predstavlja evklidsko razdaljo med vektorjema značilk f_i in g_j, ki imata vsak po 39 elementov, torej:

$$\|f_i - g_j\|_2 = \sqrt{\sum_{n=1}^{39} (f_{in} - h_{jn})^2}$$

Naloga 3 (1 točka)

V materialu za izvedbo vaje se nahaja zbirka posnetkov kratkih govornih ukazov. Gre za posnetke ukazov “gor”, “dol”, “levo” in “desno”, kjer je vsak ukaz zastopan s tremi posnetki. Z implementiranimi funkcijami za izračun MFCC značilk in DTW razdalje izvedite preizkus razvrščanja s pravilom najbližjih sosedov. Preizkus naj sledi shemi “leave-one-out”, kjer se vsak posnetek primerja z vsemi ostalimi, nato pa se ukaz razpozna na podlagi tega, kateri posnetek mu je najbližji. Na voljo je torej dvanajst posnetkov, od katerih vsakega enkrat uporabite kot testni posenetek, vse ostale pa kot prototipe. Za en testni posnetek s pomočjo postopka DTW tako pridobite enajst razdalj (cen primerjave oz. 1/podobnostjo), kot rezultat razpoznavanja pa izberete prototip, ki ustreza najmanjši vrednosti (tj., najmanjša razdalja/cena oz. največja podobnost). Ker v tem scenariju testiranja poznate pravilni odgovor razpznavanja, lahko ocenite ali je bilo razpoznavanje za dani testni posnetek pravilno ali ne. Če postopek ponovite in v vsaki iteraciji enega od dvanajstih posnetkov uporabite kot testni (tj. neznani) posnetek, lahko ocenite kakšna je uspošnost razpoznavanja s takšnim pristopom. Kot rezultat tega dela laboratorijske vaje za vsak posnetek podajte, kot kateri ukaz je bil razpoznan, poleg tega pa izračunajte in podajte še odstotek pravilno razpoznanih ukazov (odstotek pravilno razpoznanih poizkusov pri dvanajstih testih).

Naloga 4 (1 točka)

Posnemite nekaj (vsaj 4) primere svoje izgovarjave ukazov “gor”, “dol”, “levo” in “desno”. Izvedite preizkus razpoznavanje govornih ukazov po istem postopku kot pri prejšnji nalogi, le, da tokrat vse priložene posnetke, ki ste jih dobili v materialih vaje, obravnavate kot prototipe, vse lastne posnetke pa kot testne vzorce. Za vsakega od svojih vzorcev poročajte, kot kateri ukaz je bil razpoznan in podajte uspešnost razpoznavanja na vaših posnetkih. Izvedite še obratni eksperiment razpoznavanja, torej, tako, da vaši posnetki predstavljajo prototipe in priloženi posnetki predstavljajo testne vzorce, in primerjajte rezultate.

Dodatni nasveti in navodila za izvedbo

- Za zlaganje osnovnih značilk ter dinamičnih značilk prvega in drugega reda v skupni vektor značilk preučite in uporabite funkcijo np.concatenate.
- Dinamične značilke 1. reda so značilke, ki kažejo časovne spremembe osnovnih značilk. Dinamične značilke 2. reda kažejo časovne spremembe dinamičnih značilk 1. reda.
- Pri lastnih posnetkih pazite, da se karakteristike posnetkov vaših ukazov, kot so frekvenca vzorčenja in podatkovni tip zapisa, ujemajo z lastnostmi prototipnih posnetkov govornih ukazov.
- Pravilo razpoznavanja najbližjih sosedov pravi, da vzorcu, ki ga želimo razpoznati, pripišemo razred tistega prototipnega razreda, ki mu je po izbrani meri razdalje najbližji.