Εργασία στο μάθημα
Επεξεργασία Φωνής και Ήχου

Φοιτητές

Ανδρέας Κοζαδίνος Π13054
Βασίλης Τσόλης Π13167
Τηλέμαχος Καλαματάς Π14058

Θέμα

Σε αυτή την εργασία αναπτύχθηκε ένα σύστημα αναγνώρισης φωνής για αναγνώριση προτάσεων αποτελούμενων από ψηφία που έχουν ειπωθεί με αρκούντως μεγάλα διαστήματα παύσης. Η υλοποίηση έγινε σε περιβάλλον Matlab και για την εκπαίδευση καθώς και την αξιολόγηση του συστήματος χρησιμοποιήθηκαν προηχογραφημένα ψηφία από το πανεπιστήμιο της Columbia.

Εισαγωγή

Για την εργασία χρησιμοποιήθηκε η άσκηση 10.4 του βιβλίου Ψηφιακή Επεξεργασία Φωνής (Lawrence R. Rabiner, Ronald W. Schafer) για την προεπεξεργασία και κατάτμηση. Για την εξαγωγή χαρακτηριστικών χρησιμοποιήθηκαν Weighted MFCC coefficients και για την αναγνώριση ο αλγόριθμος Dynamic Time Warping Sakoe. Ακολουθεί αναλυτική περιγραφή για κάθε ένα από αυτά τα στάδια καθώς και αποτίμηση του συστήματος.

Οδηγίες Εκτέλεσης

Το αρχείο sequencerecognition.m είναι το κεντρικό αρχείο το οποίο χρησιμοποιείται για την αναγνώριση της ακολουθίας. Εκ των προτέρων είναι προγραμματισμένο να διαβάζει το αρχείο sequence.wav και να αποθηκεύει το αποτέλεσμα της αναγνώρισης στην μεταβλητή sequence. Αλλάζοντας το όρισμα στην εντολή audioread μπορεί να δοκιμαστεί κάποιο άλλο αρχείο ήχου.

Ο φάκελος Trained περιέχει τα διανύσματα χαρακτηριστικών που έχουν εκτιμηθεί κατά την εκπαίδευση αποθηκευμένα σε αρχεία .mat. Βγάζοντας από τα σχόλια την εντολή modeltraining() στην αρχή του αρχείου sequencerecognition.m μπορούμε να κάνουμε εκ νέου εκπαίδευση. Η λειτουργία modeltraining() που βρίσκεται στον φάκελο Training διαβάζει όλα τα .wav αρχεία του ίδιου φακέλου εξάγει τα διανύσματα χαρακτηριστικών και τα αποθηκεύει στον φάκελο Trained σε ξεχωριστά αρχεία.

Τέλος υπάρχει και το αρχείο evaluate.m υπεύθυνο για την αποτίμηση του συστήματος που διαβάζει όλα τα wav αρχεία από το φάκελο Samples τα αναγνωρίζει και φτιάχνει ένα confusion matrix με τα αποτελέσματα που το αποθηκεύει στην μεταβλητή matrix.

I Προεπεξεργασία

i) Αρχικά το σήμα περνιέται από ένα φίλτρο προέμφασης για να ενισχυθούν οι υψηλές συχνότητες.

```
%% preemphasis
B = [1,-0.95];
speech = filter(B,1,speech,[],2);
```

ii) Έπειτα κανονικοποιούμε τα δείγματα του ήχου ώστε να έχουν εύρος τιμών [-1,1]

```
%% normalize data
speechMin=min(speech);
speechMax=max(speech);
speech=speech/max(speechMax,-speechMin);
```

iii) Υποδειγματοληπτούμε το σήμα για ευκολία στην επεξεργασία με νέα συχνότητα Fs = 8.000 η οποία προκύπτει από τον κανόνα του Nyquist τα σήματα φωνής έχουν εύρος 300-3.400

```
%% resample input signal
x=resample(speech,Fs,FsOrig);
```

iv) Περνάμε το σήμα από ανωπερατό φίλτρο για την απομάκρυνση θορύβου χαμηλής συχνότητας

```
% highpass filtering
% Band reject 0-100Hz
% Band transition 100-200Hz
% Bandpass 100-4000Hz
hpfilter=firpm(hpforder,[0 lowcut highcut Fs/2]/(Fs/2),[0 0 1 1]);
y=filter(hpfilter,1,x);
```

II Κατάτμηση

Για την κατάτμηση χρησιμοποιήθηκε η Ενέργεια και το ZeroCrossingRate βραχέως χρόνου σε παράθυρα μήκους 30ms και frame shift 10ms. Ένα πρώτο πέρασμα γίνεται στο αρχείο digitseperation.m όπου υπολογίζονται χονδρικά οι τιμές που διαχωρίζουν τα ψηφία και έπειτα μέ ένα δεύτερο πέρασμα από το αρχείο endpoints βρίσκουμε με ακρίβεια την αρχή και το τέλος κάθε ψηφίου.

i) Υπολογισμός Ενέργειας βραχέος χρόνου και ρυθμού διέλευσης από το μηδέν.

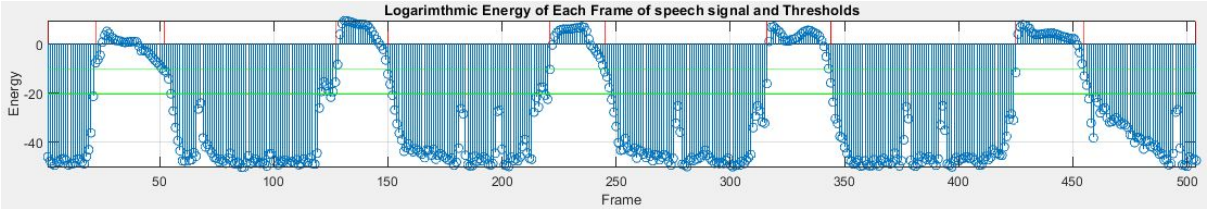
```
%% Calculate logarithmic energy and zero crossing rate for every frame
totalSamples=length(y);
preall = ceil((length(y)-L)/R); % calculate length
energy=zeros(1,preall); % memory preallocation for energy
zerocrossings=zeros(1,preall); % memory preallocation for zerocrossings
ss = 1;
count=1;
% retrieve frames from speech signal y
while (ss+L-1 <= totalSamples)
    frame=y(ss:ss+L-1).*hamming(L);
    energy(count)=10*log10(sum(frame.^2));
    zerocrossings(count)=sum(abs(diff(sign(frame))));
    ss=ss+R;
    count =count +1;
end

totalFrames=length(energy);
zerocrossings=zerocrossings*R/(2*L);
```

ii) Υπολογίζεται σε περίοδο σιγής (10 πρώτα παράθυρα του σήματος) μέση τιμή και τυπική απόκλιση για την Ενέργεια και τον Ρυθμο διελευσης από το μηδέν και έπειτα υπολογίζονται πάνω και κάτω κατώφλι για την ενέργεια και κατώφλι ρυθμού διέλευσης.

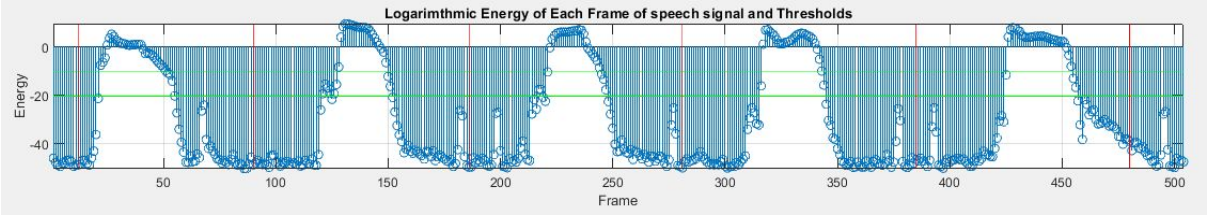
```
% Calculate average and standard deviation
% of energy and zerocrossing for background signal
% e.g first 10 frame of signal
trainingFrames=10;              %% first 10 frames
eavg=mean(energy(1:trainingFrames));
esig=std(energy(1:trainingFrames));
zcavg=mean(zerocrossings(1:trainingFrames));
zcsig=std(zerocrossings(1:trainingFrames));

%% Calculate Detection Parameters
IF=35;                          %% Constant Zero Crossing Threshold
IZCT=max(IF,zcavg+3*zcsig);     %% Variable Zero Crossing Threshold
                                %% Depends on Training
IMX=max(energy);                %% Max Log Energy
ITU=IMX-20;                     %% High Log Energy Threshold
ITL=max(eavg+3*esig, ITU-10);%% Low Log Energy Threshold
```

iii) Καλώντας την λειτουργία digitseperation βρίσκουμε στο περίπου την αρχή και το τέλος κάθε ψηφίου ακολουθόντας την λογική:

1) Βρες 20 συνεχόμενα παράθυρα που είναι κάτω από το κάτω όριο και σημείωσε αρχή σιωπής (κόκκινη γραμμή στο παρακάτω σχήμα)
2) Βρες ένα παράθυρο που είναι πάνω από το κάτω όριο
3) Κοίταξε τα επόμενα 3 παράθυρα, αν κάποιο από αυτά είναι πάνω από το πάνω όριο σημείωσε αρχή ψηφίο, αλλιώς πίσω στο 2.
4) Επανέλαβε μέχρι το τέλος



Έπειτα υπολογίζουμε ανά δύο γραμμές το μέσο τους για να βρούμε τα διαχωριστικά ανάμεσα στα ψηφία.

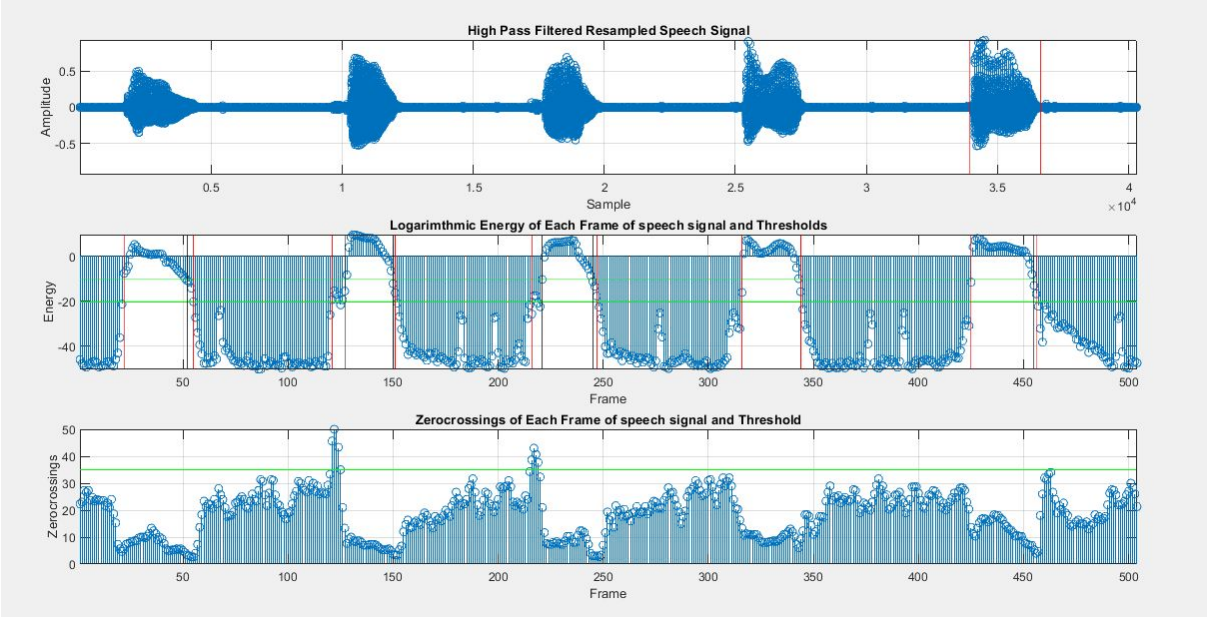


Για να βρούμε ακριβώς την αρχή και το τέλος του κάθε ψηφίου χρησιμοποιούμε την endpoints.m η οποία:

Βρίσκει την αρχή του ψηφίου:

1) Βρες ένα παράθυρο που είναι πάνω από το κάτω όριο
2) Κοίταξε τα επόμενα 3 παράθυρα, αν κάποιο από αυτά είναι πάνω από το πάνω όριο σημείωσε αρχή ψηφίου, αλλιώς πίσω στο 1.
3) Κοίταξε τα 25 προηγούμενα παράθυρα και μέτρησε πόσα από αυτά περνάνε το όριο διέλευσης από το μηδέν. Άν είναι πάνω από τέσσερα σημείωσε το τελευταίο ώς αρχή του ψηφίου.
4) Κοιταξε το προηγούμενο παράθυρο και σημείωσε το ώς αρχή του ψηφίου αν η ενέργεια του είναι μεγαλύτερη από το κάτω όριο. Επανέλαβε το 4 μέχρι η αρχή να παραμείνει σταθερή

Βρίσκει το τέλος ακολουθόντας την αντίστροφη λογική



III Εξαγωγή Χαρακτηριστικών

Η εξαγωγή χαρακτηριστικών γίνεται από το αρχείο wmfcc.m για να εξαχθούν οι συντελεστές Mel Frequency Cepstrum και να υπολογιστούν οι διαφορές delta και οι διαφορές διαφορών delta delta, χρησιμοποιήθηκε η έτοιμη συνάρτηση mfcc() με παράθυρο μήκους 25ms με ολίσθηση 10ms. Αντικαταστάθηκε ο πρώτος συντελεστής με την λογαριθμισμενη ενέργεια και στην συνέχεια υπολογίστηκε το τελικό διάνυσμα χαρακτηριστικών weighted MFCC σύμφωνα με τον τύπο:
WMFCC = MFCC + ⅓ DeltaMFCC + ⅙ DeltaDeltaMFCC
Τέλος αποθηκεύεται το διάνυσμα στο αρχείο WMFCC.mat.

```
digit = y(digitstart:digitend);
winL = 25;
winS = 10;
[WMFCC] = wmfcc(digit,Fs,winS,winL);
```

```
L = winL*Fs/1000; %% Frame Duration in Samples
R = winS*Fs/1000; %% Frame Shift in Samples
E = ((length(Digit)-L)/R)+1;
[coeffs,delta,deltaDelta] = mfcc(Digit,Fs,'LogEnergy','Replace','WindowLength',L,'OverlapLength',L-R);
[WMFCC] = coeffs + 1/3 * delta + 1/6 * deltaDelta;
```

IV Αναγνώριση

Για την αναγνώριση χρησιμοποιήθηκε Dynamic Time Warping από έτοιμο αλγόριθμο από τα έγγραφα του μαθήματος Αναγνώριση Προτύπων. Αρχικά διαβάζονται όλα τα .mat αρχεία από τον φάκελο Trained που περιέχουν τα διανύσματα χαρακτηριστικών από την εκπαίδευση και το αρχείο WMFCC.mat που περιέχει το διάνυσμα χαρακτηριστικών του ψηφίου προς αναγνώριση. Έπειτα υπολογίζεται το κόστος DTW για κάθε ένα από τα ψηφία του συνόλου εκπαίδευσης και επιλέγεται αυτό με το μικρότερο κόστος. Η διαδικασία επαναλαμβάνεται για όλα τα ψηφία της ακολουθίας και τα αποτελέσματα σώζονται στην μεταβλητή sequence.

```
sequence =

     1     2     3     4     5
```

Εκπαίδευση

Η εκπαίδευση γίνεται από το αρχείο modeltraining.m το οποίο εκτιμά τα διανύσματα χαρακτηριστικών από τα αρχεία ψηφίων που βρίσκονται στο φάκελο Training περνώντας από όλα τα στάδια που περιγράφηκαν παραπάνω και αποθηκεύει τα αποτελέσματα σε ξεχωριστά αρχεία στο φάκελο Trained. Τα αποτελέσματα είναι labeled με το όνομα τους. Η διαδικασία της εκπαίδευσης διαρκεί περίπου 5 λεπτά.

Αποτίμηση Του Συστήματος

Για την αποτίμηση του συστήματος χρησιμοποιήθηκε το αρχείο evaluation.m το οποίο διαβάζει όλα τα αρχεία Samples και τα ταξινομεί σύμφωνα με την διαδικασία που περιγράφηκε παραπάνω. Δοκιμάστηκαν μερικές παραλλαγές με διαφορετικά αποτελέσματα. Κρατήθηκε η τέταρτη που είχε το πιο ψηλό ποσοστό

1) Δοκιμή πρώτη (ποσοστό επιτυχίας 86.66%)
    a) 40 Αρχεία εκπαίδευσης 2 ομιλητές από δύο φορες ο καθένας το κάθε ψηφίο
    b) 180 testing αρχεία
    c) Διάνυσμα Χαρακτηριστικών μόνο WMFCC

	0	1	2	3	4	5	6	7	8	9
0	18	0	0	0	1	0	0	0	0	1
1	0	18	0	0	1	4	0	2	0	5
2	0	0	18	0	0	0	0	0	0	0
3	0	0	0	18	0	0	0	0	0	0
4	0	0	0	0	16	0	0	0	0	0
5	0	0	0	0	0	13	1	1	0	0
6	0	0	0	0	0	0	14	0	3	0
7	0	0	0	0	0	1	1	14	0	0
8	0	0	0	0	0	0	2	0	15	0
9	0	0	0	0	0	0	0	1	0	12

2) Δευτερη δοκιμή (ποσοστό επιτυχίας 82.22%)
    a) 40 Αρχεία εκπαίδευσης 2 ομιλητές από δύο φορες ο καθένας το κάθε ψηφίο
    b) 180 testing αρχεία
    c) Διάνυσμα Χαρακτηριστικών WMFCC με την λογαριθμισμενη ενέργεια αντί του πρώτου συντελεστή

	0	1	2	3	4	5	6	7	8	9
0	18	0	0	0	2	0	0	0	0	0
1	0	18	0	0	0	2	0	2	0	5
2	0	0	18	0	0	0	0	0	0	0
3	0	0	0	18	0	0	0	0	0	0
4	0	0	0	0	16	0	0	0	0	1
5	0	0	0	0	0	15	0	0	0	0
6	0	0	0	0	0	0	14	0	2	0
7	0	0	0	0	0	0	1	14	0	1
8	0	0	0	0	0	0	3	0	16	0
9	0	0	0	0	0	1	0	2	0	11

3) Τρίτη δοκιμή (ποσοστό επιτυχίας 83%)
    a) 40 Αρχεία εκπαίδευσης 2 ομιλητές από δύο φορες ο καθένας το κάθε ψηφίο
    b) 180 testing αρχεία
    c) Διάνυσμα Χαρακτηριστικών WMFCC με την λογαριθμισμενη ενέργεια αντί του πρώτου συντελεστή
    d) Επιλέγεται το ψηφίο που έχει το ελάχιστο μέσο κόστος από όλους του ομιλητές.

	0	1	2	3	4	5	6	7	8	9
0	18	0	0	0	0	0	0	0	0	0
1	0	17	0	0	0	4	0	2	0	5
2	0	0	17	0	0	0	0	0	0	0
3	0	0	0	18	0	0	0	0	0	0
4	0	0	0	0	17	1	0	0	0	1
5	0	0	0	0	0	10	0	0	0	0
6	0	0	0	0	0	0	14	0	3	0
7	0	1	1	0	1	2	1	14	0	1
8	0	0	0	0	0	0	3	0	15	0
9	0	0	0	0	0	1	0	2	0	11

4) Τέταρτη Δοκιμή (ποσοστό επιτυχίας 87.85%)
    a) 80 Αρχεία εκπαίδευσης 2 ομιλητές από δύο φορες ο καθένας το κάθε ψηφίο
    b) 140 testing αρχεία
    c) Διάνυσμα Χαρακτηριστικών WMFCC με την λογαριθμισμενη ενέργεια αντί του πρώτου συντελεστη

	0	1	2	3	4	5	6	7	8	9
0	14	0	0	0	0	0	0	0	0	0
1	0	13	0	0	0	0	0	1	0	1
2	0	0	14	0	0	0	0	0	0	0
3	0	0	0	12	0	0	0	0	0	0
4	0	0	0	0	13	0	0	0	0	0
5	0	1	0	0	1	14	0	1	0	2
6	0	0	0	0	0	0	11	0	2	0
7	0	0	0	0	0	0	0	10	0	1
8	0	0	0	2	0	0	3	0	12	0
9	0	0	0	0	0	0	0	2	0	10

5) Πέμπτη Δοκιμή (ποσοστό επιτυχίας 85%)
   a) 80 Αρχεία εκπαίδευσης 2 ομιλητές από δύο φορες ο καθένας το κάθε ψηφίο
   b) 140 testing αρχεία
   c) Διάνυσμα Χαρακτηριστικών WMFCC μόνο

	0	1	2	3	4	5	6	7	8	9
0	13	0	0	0	0	0	0	0	0	0
1	0	13	0	0	0	0	0	2	1	0
2	0	0	10	0	0	0	0	0	0	1
3	0	0	0	12	0	0	0	0	0	0
4	0	0	1	0	14	0	0	0	0	0
5	0	1	0	0	0	14	1	0	1	2
6	0	0	0	0	0	0	10	0	0	0
7	1	0	1	0	0	0	0	10	0	0
8	0	0	2	2	0	0	3	0	12	0
9	0	0	0	0	0	0	0	2	0	11